Feasibility of an Electric Jetpack

A thesis

submitted in partial fulfilment

of the requirements for the Degree

of

Master of Engineering in Electrical and Electronic Engineering

in the

University of Canterbury

By

Timothy John Youard

University of Canterbury

2010

ABSTRACT

The Martin Aircraft Company Limited has been developing the Martin Jetpack for over 25 years. The recent worldwide launch of the Jetpack has enabled the company to step up its research and development programme. The goal of this project was to determine the feasibility of an electrically powered version of the Martin Jetpack. The feasibility of the Electric Jetpack was determined by researching energy storage technologies, researching power cable technologies, simulations of flight times, surveys of electric motors, and the development of a simulation program which was used to optimise some preliminary custom motor designs. The overall conclusion of this project was that the Electric Jetpack was feasible only when it was powered through a tethered power cable, and on-board energy storage was not used.

An investigation into current energy storage technologies showed that the Electric Jetpack is not considered feasible when using on-board energy storage, however it is possible to obtain flight for a very short time. The energy storage technologies studied were batteries, fuel cells, and ultra-capacitors. It was found that the best performing technology was the lithium iron nano-phosphate battery. A simulation of flight time showed that this battery type would be able to provide flight for approximately 3.6 minutes. Future trends indicated that the Electric Jetpack with on-board energy storage may eventually be feasible when using a lithium-ion based battery due to improvements being made in energy density and power density.

By using a tethered power cable, the weight of the on-board energy storage could be eliminated. This was shown to be a feasible method for powering the Electric Jetpack for applications where the Jetpack needs to only be operated in a small area. The best cable type to use was a multi-stranded flexible cable operating at a high DC bus voltage. The weight of a 5 meter power cable using a 1000 V bus voltage was shown to be 4.9 kg. Potential applications for this kind of Jetpack could include thrill rides and rescue operations from multi-storied buildings. A cable made from carbon nanotubes was shown to be a future technology that could offer a lighter cable.

A survey of currently available electric motors showed that none met both the power density and speed required by the Electric Jetpack, even when using a tethered power cable to eliminate the energy storage weight. Because of this, a custom motor design was needed. Research into motor technologies showed that the permanent magnet brushless DC (PMBLDC) motor was the most suited type for the Electric Jetpack. The permanent magnet brushless AC (PMBLAC) motor was also suitable. A PMBLDC motor simulation program was developed using MATLAB which could be used to optimise preliminary custom designs. A characterisation of allowable motor time constants for the PMBLDC motor type was made in order to speed up the simulation time.

The optimisation results showed that a power density of 5.41 kW/kg was achievable for the motor when it was located inside the ducted fan tubes, and a power density of 6.56 kW/kg was achievable when the motor was located outside the ducted fans and operated at a higher speed. The motor designs were shown to be within the expected torque per unit rotor volume (TRV) range for aerospace machines. The best power density figures would leave between 37 kg and 42 kg of weight for the motor driver/controller, cable weight, and miscellaneous motor parts. This was considered to be feasible. An FEM simulation was made on one of the optimised motor designs. The FEM results agreed with the parametric results within reasonable accuracy. The parametric back-EMF waveform over-estimated the effects of slotting.

ACKNOWLEDGEMENTS

Firstly I would like to thank my supervisor Dr. Paul Gaynor and associate supervisor Professor Philip Bones for guidance and assistance during this project.

I would like to thank the staff in the Department who were friendly and helped out when I had questions.

I would like to thank the Foundation for Research, Science and Technology for providing financial assistance in the form of a TIF grant.

I would like to thank Glenn Martin and the Martin Aircraft Company Limited for providing the opportunity to work on this novel project.

And finally I would like to thank my parents for supporting me while I studied.

CONTENTS

LIST OF FIGURES

LIST OF TABLES

Chapter 1

Introduction

1.1 Background

The Martin Jetpack is a combustion engine powered jetpack which uses two ducted fans to provide the lift required for flight. Figure 1.1 shows a photograph of the Jetpack in operation. The Martin Jetpack is the result of over 25 years of research and development. The recent worldwide launch of the Jetpack has enabled Martin Aircraft Company Limited to step-up its research and development programme. This has lead to the proposal of the Electric Jetpack.

Figure 1.1: The Martin Jetpack [1].

The main advantage of using an electric motor instead of a combustion engine to power the ducted fans is that the noise level can be reduced. For an example, it has recently been suggested that all electric and hybrid vehicles include fake engine sounds in order to increase safety, due to the low noise levels that electric motors offer [2]. The current Jetpack uses a 150 kW combustion engine [3], and is very loud (a measurement of sound level during takeoff was 85 dB at 50 meters away). If an electric motor is used, then there is the potential to expand the possible markets of

the Jetpack to include urban areas where excessive noise is a problem. Less noise also makes the Jetpack less intimidating and may attract more people to fly the Jetpack.

Another advantage of using an electric motor is that no greenhouse gasses or pollution is emitted. This would help to attract the environmentally concerned market. Another advantage is that that an Electric Jetpack opens up manoeuvrability and control advantages. If an electric motor is contained in each duct, then each duct can be driven at a different speed and produce a different lift.

The current Jetpack engine weighs only 60 kg [3], and this would need to be replaced with an electric drive system. It was unclear whether the Electric Jetpack is a feasible idea from an engineering perspective. A feasibility study is the first step to determine whether pursuing the Electric Jetpack is worthwhile. The feasibility study here focuses on the engineering problem, and includes many different topics. These include the current and future state of energy storage technologies, a survey of electric motors, and in this case, the simulated design of some preliminary custom motors.

1.2 Goals for the Project

Since this was a feasibility study, the goals of the project needed to adapt, depending on the results that were found as the project proceeded. The objectives progressed as follows:

- Determination of the electric motor requirements:

  Characterise the torque-speed requirements. Determine possible configuration options. Determine the power density requirements. Determine any other requirements.

- Research motor technology:

  Determine the most suitable motor type to power the Electric Jetpack.

- Survey existing electric motors to find an off-the-shelf system:

  The Survey needed to include all parameters relevant to the requirements of the Electric Jetpack as determined from the previous objective. This survey was used to determine whether a suitable off-the-shelf system could be found.

- Research current and future trends in energy storage technology:

  Battery, fuel cell, and ultra-capacitor technologies were researched. The results from this determined whether the Electric Jetpack using on-board energy storage was feasible.

- Research power cable technology:

  Research whether a tethered power cable Electric Jetpack is feasible. The energy to power the Jetpack is supplied through the cable.

- Simulate and optimise possible motor designs.

  This is to determine whether an electric motor can be built which meets the requirements for the Electric Jetpack.

1.3 Thesis Overview

Chapter 2 deals with the motor requirements and the possible configuration options for the motor. The torque-speed requirements are analysed. An approximation of the power density required for the motor is made. The dimensional restrictions of the motor are determined and the minimum efficiency desired is also discussed.

Chapter 3 presents some research into different motor types in order to determine the most suited type for the Electric Jetpack. Some introductory theory behind the operation of permanent magnet brushless synchronous motors is presented. An investigation into the motor cooling is made and a survey of motors is presented. The survey is used to determine whether there are any off-the-shelf motors that can be used.

Chapter 4 investigates energy storage technologies. Different technologies are investigated and compared in order to find the most suitable energy storage technology for the Electric Jetpack. Simulations of flight times are made to determine the flight time when using the lithium iron nano-phosphate battery and the ultra-capacitor. This allows the feasibility of an Electric Jetpack using on-board energy storage to be determined.

Chapter 5 determines the best type of cable to use for a tethered Electric Jetpack. An investigation into various material options for the cable is presented. The most suitable cable type is determined and some calculations of the required cable weight are made. Some possible future cable technologies are discussed. Possible applications for the tethered Electric Jetpack are also discussed.

Chapter 6 details the theory behind a simulation program which was made to simulate permanent magnet brushless DC (PMBLDC) motors. The simulation program was written so that it may be used to simulate preliminary custom motor designs for the Electric Jetpack. An investigation into the motor time constant and its effects on torque is also made.

Chapter 7 gives the results for some preliminary motor designs that were simulated using the PMBLDC simulation program that was written using MATLAB. The motor parameters used to produce high power density designs are discussed. The results are presented for a range of motor variations. These include inside duct and outside duct configurations, inside and outside-rotor variations, a range of magnet pole counts, and different lamination materials. A discussion of the results is presented.

Chapter 8 compares an FEM simulation with one of the simulation results from the optimised motor designs. The comparison is used to validate that the simulation program is working properly, and to identify any areas which could be improved.

Chapter 9 presents the project conclusions, and discusses future work that should be done if the Electric Jetpack project is further pursued.

Chapter 2

Motor Requirements

2.1 Introduction

The Electric Jetpack has a number of requirements that must be satisfied in order to make it feasible. The most important factors to consider are the power density and the torque-speed requirements, since these two aspects determine whether the Electric Jetpack can obtain flight. Other requirements to consider are the dimensional restrictions and motor efficiency. This chapter investigates these requirements so that an investigation into different motor types can follow.

2.2 Possible Motor Configurations

There are two configurations which can be used to drive the ducted fans. The first is to have a motor placed directly within each duct. The second is to have a single motor coupled to both of the ducted fans via the existing belt drive. Both of these configurations are considered to be viable options.

The advantages for an inside duct configuration are:

- The fan airflow may be used to provide cooling assistance to the motor.
- Independent control of each fan leads to potential manoeuvrability advantages.
- Uses the existing space within each fan duct, allowing a more compact design.
- Direct coupling of the motors to the fan eliminates the belt drive losses (≈2 %).
- The weight of the belts and associated gear wheels is eliminated.

The advantages for an outside duct configuration are:

- The dimensional requirements of the motor are not restricted.
- The motor is simpler to fit to the existing Jetpack design.

- A higher speed motor can be used and geared down using a 1:0.85 ratio belt drive. Higher speed motors allow reduced dimensions and an increase in power density [4].

2.3 Dimensional Requirements

If the electric motor is located within the ducts, it must fit within the space provided in the fan duct boss tube. Figure 2.1 shows the main dimensions of the tube. Ideally the diameter should not be changed, however the length of the tube can be extended as necessary as this does not obstruct the airflow. The space enclosed by the dome on the bottom of the tube can also be utilised.

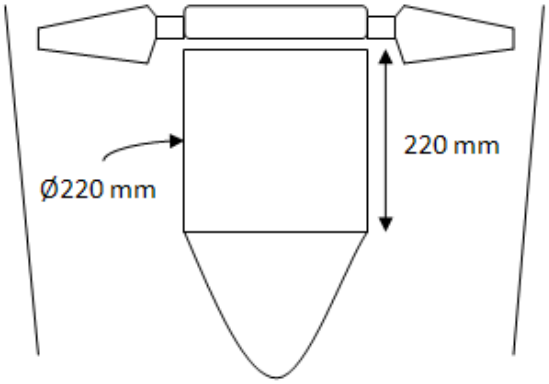


Figure 2.1: Fan duct boss tube dimensions.

2.4 Weight and Power Density Requirements

Table 2.1 shows the weights of the Jetpack parts and the max pilot/load weight.

Weight (kg)	
Max pilot/load	127
V4 engine	60
Fuel tank	5
Cooling system	8
Empty weight	113
Safety equipment	3

Table 2.1:Weights for the Martin Jetpack [3].

The existing combustion engine weighs approximately 60 kg (power density of 2.5 kW/kg), and this can be used as an approximate goal power density for a replacement electrical drive system. This must include the electric motor(s), motor driver, and any on-board energy storage. Removal of the fuel tank allows an extra 5 kg in weight savings. The cooling system is not removed as it is used to allow liquid cooling of the electric motor(s). If the Electric Jetpack is powered through a tethered power cable, then the weight of the on-board energy storage can be eliminated and the weight of the power cable must be added. The weight of the power cable depends on the desired length and voltage used. An exact value for the minimum required power density for the electric motor cannot be determined at this stage in the investigation.

As an example, if the combined electric drive system obtained a weight of 80 kg, which is 15 kg heavier than the existing combustion engine and fuel tank, then the Electric Jetpack could still be feasible, but it could only safely carry a 112 kg person/load. For these reasons, a specific value for the power density requirement for the electric motor has not been set. Any motors surveyed which approach the power density of the combustion engine (2.5 kW/kg) are evaluated on a case by case basis to determine what useful pilot/load weight they can support.

2.5 Torque-Speed Requirements

On the combustion engine based Jetpack, the ducted fans operate at a torque of 85 Nm while at a rotational speed of 7000 RPM [3]. This operating point is used as the desired operating point for the Electric Jetpack, as the weight will not differ significantly. The Electric Jetpack cannot be much heavier due to the fan speed limitations. If the fan speed is limited to 7000 RPM, then the thrust each fan produces is also limited. This in turn limits the maximum weight of the Electric Jetpack to be the same as the existing combustion engine based Jetpack. The Electric Jetpack is also unlikely to be significantly lighter if a battery is used as the power source. Cells will be added to the battery until the Electric Jetpack weighs the same as the combustion engine based Jetpack, so that the flight time can be maximised.

The motor should be able to produce a torque which is 20% higher than fan operating point torque over the entire rotational speed range of the fan. This was determined from consultation with Martin Aircraft. Some reasons for this are to make the Jetpack responsive, safe, and to ensure it operates under various load factors. Varying load factors are caused by different pilot weights and air densities. Since the existing combustion engine based Jetpack operates at a nominal speed of 7000 RPM, and at a torque of 85 Nm for each fan (when carrying the rated pilot/load), the motor used must produce at least 17 Nm of extra torque over the entire range of rotational speeds.

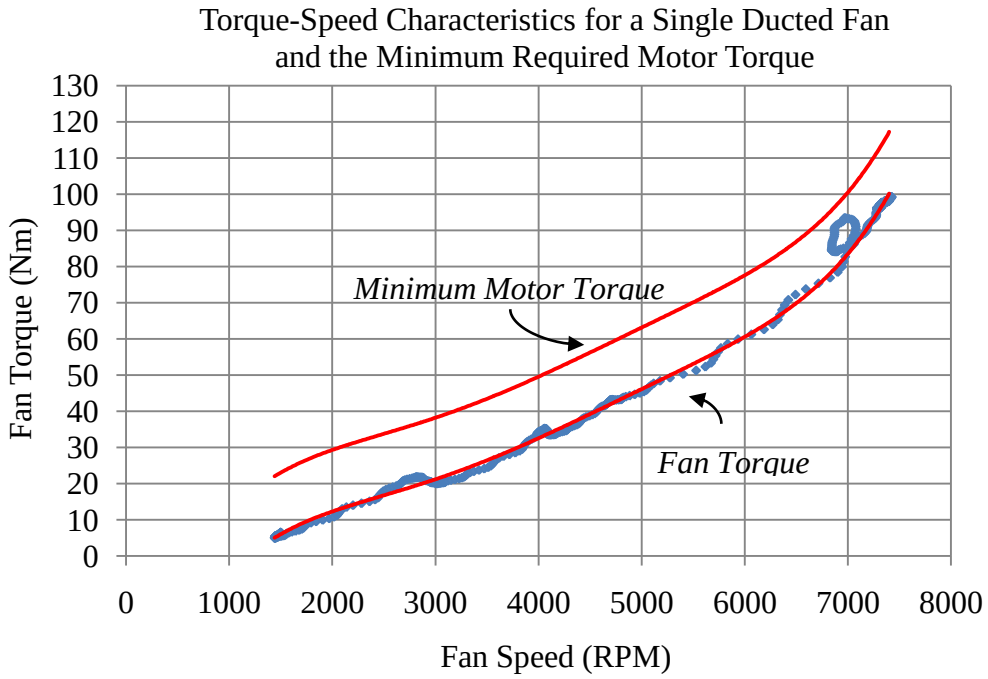


Figure 2.2: Characteristics for a single ducted fan.

Figure 2.2 shows the torque-speed characteristics for a single ducted fan, with the minimum required motor torque plotted also. Fans follow an approximately quadratic torque-speed relationship. In this case, a polynomial of order five was used to fit the data more precisely. In Figure 2.2, and in subsequent figures, the data points form a loop at approximately 7000 RPM. This data was not used to fit the polynomials due to its double valued nature. The most likely reason the loop occurred is because the motor was controlled by a person. The person operating the accelerometer has let the speed fall down slightly before correcting this and speeding up the engine again. Control of the motor speed is very sensitive when operating the accelerometer by hand.

The data used to produce the graph in Figure 2.2 comes from a test run of a single ducted fan with the blade angle and the stators set to a default angle. Changing the blade angle changes the torque-speed characteristics slightly, however the extra 20% factor added to the motor torque helps to take adjustments into account. The same data is used to find the power vs. thrust relationship, as is shown in Figure 2.3. This relationship is used to determine the flight time of the Electric Jetpack by simulation.

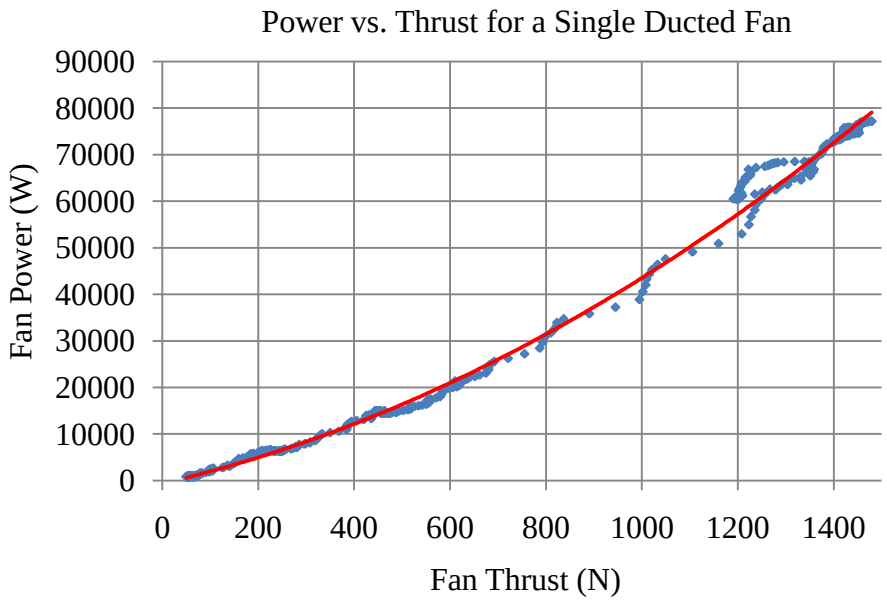


Figure 2.3: Fan power output vs. thrust for a single ducted fan.

The maximum operating point for the inside duct motor configuration is 102.3 Nm at 7000 RPM. The corresponding power output is 75 kW per fan. The outside duct configuration is passed through a belt drive with a 1:0.85 ratio, and hence the maximum operating point is 173.9 Nm at 8235 RPM. The corresponding power output is 150 kW.

2.6 Motor Efficiency Requirements

The efficiency of the motor used is important in this application, as this will have a direct effect on the possible flight time of the Electric Jetpack when using on-board energy storage. The other reasons for requiring an efficient motor is to reduce the cooling requirements of the motor, and to reduce the size, weight, and power requirements of the motor driver. For this application any motor with an efficiency of 90% or greater at the desired operating point is considered.

2.7 Summary

The requirements for the electric motor have been determined. A good target power density for the electric drive system is around 2.5 kW/kg, however this can be made smaller if the pilot/load weight is reduced. When the outside duct configuration is used, the motor speed is 8235 RPM and the power required is 150 kW. For the inside duct configuration the motor speed is 7000 RPM and the power required is 75 kW (per fan). The dimensions of the duct boss tube create dimensional restrictions for the inside duct configuration. The outside duct configuration may allow an increase in power density due to the motor operating at a higher speed.

Chapter 3

Motor Technology Investigation

3.1 Introduction

In order to determine the feasibility of the Electric Jetpack, the current state of electric motor technology was investigated. This chapter details this investigation from the point of view of satisfying the requirements set out in Chapter 2. An investigation into liquid cooling is also presented. The results of a survey of electric motors used in electric traction applications and small electric aircraft is also presented in order to determine whether an off-the-shelf motor could be found to power the Electric Jetpack, or whether a custom designed motor should be used.

3.2 Choice of Motor Technology

Brushless motors are the best type to use in the Electric Jetpack. This is because brushless motors are smaller, have higher efficiencies [5], and are more reliable than brushed motors. The use of brushes for commutation leads to high maintenance requirements and/or a short lifetime [6]. Three types of brushless motors are considered here. They are the permanent magnet brushless DC or AC motor (PMBLDC/AC), the AC induction motor (IM), and the switched reluctance motor (SRM). PMBLDC motors are often called brushless DC motors (BLDC) for short.

Table 3.1 gives a comparison between four different motor types; the brushed DC series wound motor, the induction motor, the PMBLDC/AC motor, and the switched reluctance motor. This comparison was based on power levels required for traction motors to be used in electric and hybrid vehicles, so it is valid for the power levels associated with the Electric Jetpack. It can be seen that PMBLDC/AC motors have the highest power density, while induction motors and switched reluctance motors fall behind, and have roughly equal power densities. The other advantages of the PMBLDC/AC motor are its exceptional efficiency and high reliability. However, the

use of permanent magnet material makes this motor type a more expensive option [7].

Characteristics	**Motor Type**			
	DC	**IM**	**PMBL DC/AC**	**SRM**
Power Density	2.5	3.5	5	3.5
Efficiency	2.5	3.5	5	3.5
Controllability	5	5	4	3
Reliability	3	5	4	5
Technological Maturity	5	5	4	4
Cost	4	5	3	4
Total Score	22	27	25	23

Table 3.1: Comparison between different motor types. Scored from one (lowest rating) to five (best rating) [5].

Induction motors, switched reluctance motors, and permanent magnet brushless motors can all be designed to operate at greater than 90% efficiency. For induction motors, it is quite possible to obtain an efficiency of 94% or greater at 75 kW. As an example, "National Electrical Manufacturers Association" (NEMA) standards for premium efficiency, 3600 RPM, totally enclosed, fan cooled induction motors is set at 94.1% at the 75 kW level [6]. Table 3.1 shows that the SRM has a roughly equal efficiency compared to the induction motor. PMBLDC/AC motors can obtain efficiencies greater than induction motors due to the lack of resistive losses in the rotor. The rotors use permanent magnets to create a magnetic field, as opposed to induction motors which rely on induced currents in the rotor conductors. It is not uncommon to see efficiencies above 95% for the PMBLDC/AC motors. Examples can be found in the motor survey which is presented in Appendix A.

The Jetpack is a near constant high speed application. The torque ripple of the motor is not too important due to the inertia of the ducted fans at high speed. The current combustion engine exhibits a torque ripple peak-peak of around 60% of the average torque at rated speed. Because of this, any of the motors discussed above are suitable in terms of meeting torque ripple/quality requirements. Given this fact, and the discussions about power density and motor efficiency above, the most suited motor technology for the Electric Jetpack is the PMBLDC/AC motor.

3.3 Permanent Magnet Brushless Motor Basics

This section gives a basic overview of PMBL (DC or AC) motor terminology and operation that is needed to understand the rest of this report. In-depth understanding of PMBL motor operation and theory can be obtained from relevant texts [8-10].

The PMBL motor consists of two primary parts. These are the rotor and the stator. Figure 3.1 shows cross-sections of two examples of a PMBLDC motor with 12 slots in the stator and 4 magnet poles bonded to the rotor. The rotor can be positioned on the inside, as is shown in Figure 3.1(a), or the outside, as is shown in Figure 3.1(b). The rotor yoke and stator yoke refer to the iron laminations used to contain and direct the flux from the permanent magnets. The stator yoke has a number of slots in which copper coils are placed. The coils are connected together to form phase windings. It is typical to have three phase windings. The connections to the phase windings are made accessible on the outside of the motor so they can be energised by an external motor driver.

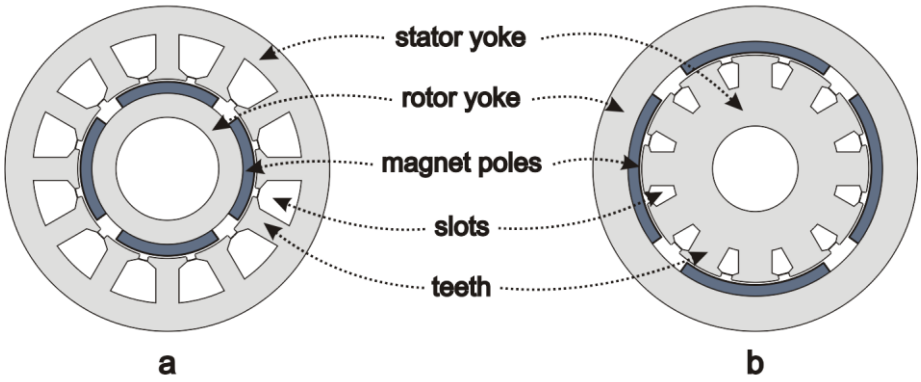


Figure 3.1: PMBLDC motor cross-sections for (a) inner-rotor and (b) outer-rotor types.

As the rotor permanent magnets rotate, the flux linked by each set of phase windings changes, which induces a back-EMF into the phase windings in proportion to the rate of change of flux linkage (by Faraday's law). The back-EMF per coil can be calculated as

$$\varepsilon = N\frac{d\emptyset}{dt} \tag{3.1}$$

where N is the number of turns, $\emptyset$ is the magnet flux through the coil, and t is time.

By forcing a current into the phase windings in synchronism with the shape of the back-EMF, a positive output torque is generated. The total back-EMF per phase is simply the sum of the back-EMFs for each coil belonging to the same phase (assuming the coils are connected in series). The back-EMF constant of a PMBL motor refers to the magnitude of the back-EMF seen at the motor terminals per unit of rotational speed. By using conservation of energy, the instantaneous torque produced per phase can be calculated as

$$\tau_p(t) = \frac{\varepsilon_p(t) i(t)}{\omega} \tag{3.2}$$

where $\varepsilon_p(t)$ is the instantaneous phase back-EMF, $i(t)$ is the instantaneous phase current, and ω is the rotational velocity of the rotor (assumed constant).

Figure 3.2 shows the equivalent circuit for a star connected three-phase PMBL motor. The delta connected motor is not as efficient as the star connected motor, as the delta configuration opens up the possibility to have circulating currents around the phase windings, which produce additional resistive losses [10]. The phase resistance, phase inductance, and phase back-EMF can also be seen in the equivalent circuit. The phase resistance occurs because copper is not a perfect conductor. The phase inductance occurs due to the self and mutual inductance of the windings, and the phase back-EMF is due to the changing magnetic flux linking the windings as described previously. To drive the desired currents into the windings, a three-phase full-bridge circuit can be used, which is also shown in Figure 3.2. The currents can be pulse width modulated to be either sinusoidal or quasi-square-wave in nature. These are described in more detail in section 3.4.

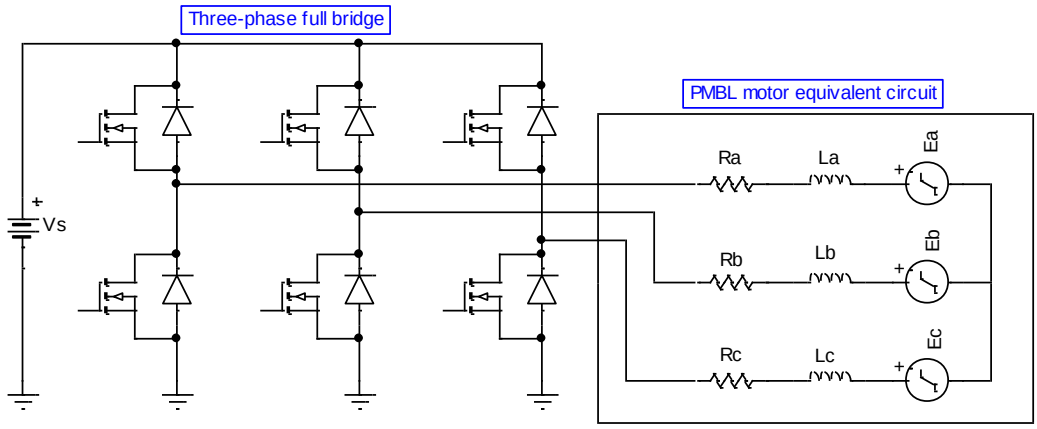


Figure 3.2: PMBL motor equivalent circuit and full-bridge driver. Symbols "a", "b", and "c" refer to the specific phase, "R" is the resistance, "L" is the inductance, "E" is the back-EMF, and "Vs" is the supply voltage.

PMBL motors are very efficient. The losses in PMBL motors can be split up into three main components: the resistive loss, the core loss, and the mechanical loss. The resistive loss is usually the largest component [10]. The resistive loss can be reduced by decreasing the current density in the windings, or by decreasing the winding temperature. The core loss is due to hysteresis and eddy current loss in the motor laminations. These losses are made worse by increasing the electrical frequency, or by increasing the magnetic flux density in the laminations. The electrical frequency is the frequency of rotation multiplied by the number of magnet pole pairs, and is equal to the fundamental frequency of the magnetic flux density waveform within the laminations. The mechanical loss is due to friction inside the bearings and from the internal air-flow resistance inside the motor.

3.4 Permanent Magnet Brushless AC vs. DC

While the PMBL motor has been shown to be most suitable for the Electric Jetpack, the choice between PMBLAC and PMBLDC remains. Both motors operate under similar principles [11]. As the rotor permanent magnets rotate, the flux linked by each set of phase coils changes, which induces a back-EMF into the phase windings in proportion to the rate of change of flux linkage. A current is forced into the phase windings in synchronism with the shape of the back-EMF in order to generate torque.

PMBLAC motors have a sinusoidal back-EMF, and are usually driven using sinusoidal currents in synchronism with the back-EMF. PMBLDC motors have a trapezoidal back-EMF, and are usually driven using quasi-square-wave currents in synchronism with the back-EMF [11]. Figure 3.3 shows a visual representation of the ideal back-EMF and currents for both motor variations. When the torque produced from all the phases is combined, the total torque is smooth with no ripple [11]. In reality, it is not possible to produce the perfect quasi-square-wave current waveform required for the PMBLDC motor. This is because each phase has self and mutual inductance, which means that the current cannot rise instantaneously. The result is that the current does not follow the ideal shape, and a torque ripple is generated at each commutation. The PMBLAC motor does not have this problem, as the desired current waveform is sinusoidal with no instantaneous changes required.

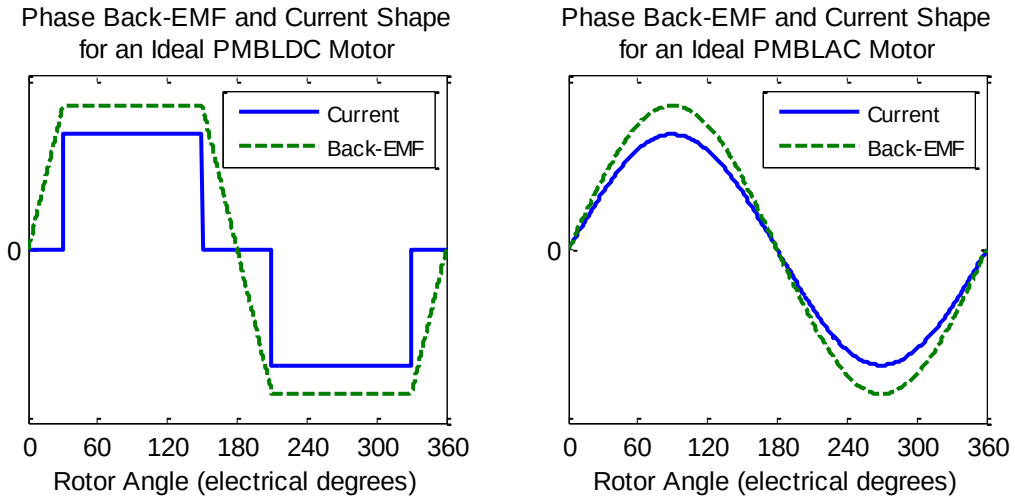


Figure 3.3: Phase Back-EMF and current shapes for the ideal PMBLDC and PMBLAC motor.

There are many different methods used to create the desired back-EMF shape. Real motors produce only approximations to the ideal back-EMF shapes. The corners on the PMBLDC back-EMF waveform are rounded rather than ideal rectangular corners due to the effect of fringing and flux leakage near the inter-polar regions [12]. In PMBLAC motors, it is possible to obtain a back-EMF close to a perfect sinusoid, depending on the method used. Figure 3.4 shows a number of different magnet shape and placement options for inside-rotor PMBL motors. The motor in Figure 3.4(a) uses radial magnetized magnets and produces an approximate trapezoidal back-EMF waveform suitable for PMBLDC motors, especially when combined with

concentrated windings. The motors shown in Figure 3.4(b-d) are more suitable for producing a sinusoidal back-EMF for PMBLAC motors, and are usually combined with sinusoidal and overlapped winding distributions [10, 11]. Skewing of the magnets or the slots can also be used to help produce a sinusoidal back-EMF. This also has the added benefit of reducing the cogging torque of the motor [13], but can also reduce torque output [10]. The radial magnetized surface mounted magnet motor shown in Figure 3.4(a) can be inverted to form an outside-rotor motor.

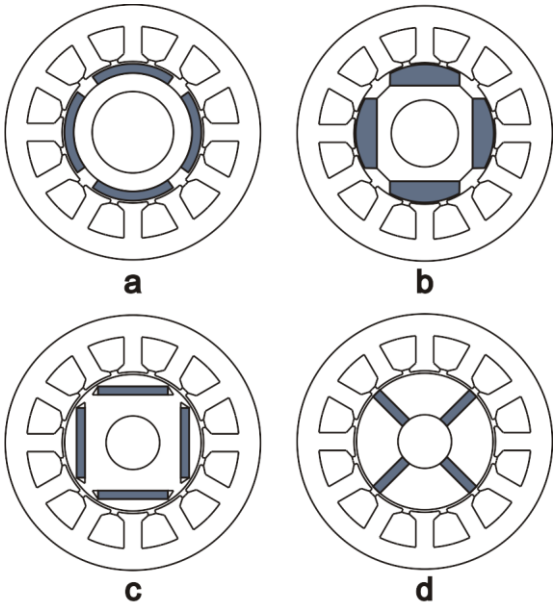


Figure 3.4: Possible rotor designs for PMBL inside-rotor motors [9, 10].

The ideal PMBLDC motor has a power density and torque output advantage over the PMBLAC motor. This advantage is quoted as 5% [11] and also 15% [8], and depends on the conditions of the comparison. The 15% figure assumes equal RMS current in the stator, and equal peak back-EMF for both the PMBLDC and AC motor variations. The 5% figure assumes equal RMS current and equal RMS back-EMF for both motor variations. It is difficult to determine which figure is more accurate. The reason different figures are being quoted may be because it is hard to relate the back-EMF shape and amplitude directly to the mass of the rotor and magnets, as there are many methods that can be used to produce the sinusoidal back-EMF (like the ones shown in Figure 3.4). As previously mentioned, the inductance of the windings means that the ideal current waveform for the PMBLDC motor cannot be

achieved. The consequence of this is that the 5-15% power density advantage of the PMBLDC motor may be reduced as the electrical frequency of the motor increases.

As discussed previously, PMBLAC motors can produce smooth torque with very little ripple, while PMBLDC motors always have some commutation torque ripple. The torque ripple is not very important for high speed fan applications such as the Electric Jetpack. A PMBLDC motor with commutation ripple would still be better than the 60% torque ripple currently experienced on the combustion engine based Jetpack. For this reason, the torque quality/ripple is not a deciding factor between PMBLAC and DC motors for this application.

One advantage of PMBLAC motors is the ability to apply field weakening control [11]. Field weakening uses a portion of the current to reduce the magnetic field from the permanent magnets, and thus decrease the back-EMF constant. This allows the speed to be increased beyond the base speed for a given supply voltage, however the torque decreases as this is a constant power region [11]. The Electric Jetpack operating point is nearly constant at 7000 RPM, and a constant power region above base speed is not needed.

The PMBLDC motor is typically easier and cheaper to manufacture. The windings are typically concentrated single-layer or double-layer variations which are easier to wind compared to the sinusoidally distributed windings used for PMBLAC motors [11]. The outer-rotor variation of the PMBLDC motor has further benefits since the windings can be wound using a conventional winding machine used to wind rotor armatures of brushed DC motors. The outer-rotor variants hold the magnets in place with the rotor yoke, without the need for a stainless steel or kevlar retaining ring that may be required in high speed inside-rotor motors. The motor driver for the PMBLDC motor is simpler, as it can be operated by six-step commutation, as opposed to the PWM sine-wave modulation used for PMBLAC motors. However, it should be noted that the sine-wave drive could also be used on the PMBLDC motor and vice-versa. The consequence of doing this is an increase in torque ripple [11].

3.5 Motor Cooling

Motor cooling is very important in regards to designing a high power density motor. A higher current density or current per slot can be used to increase motor power density, since this decreases the mass of copper and yoke iron required. The disadvantage is an increase in winding losses in each slot, as the copper area per slot is decreased. Running the motor at high speeds also increases the power density, and consequently increases the core losses. The heat from the winding and core losses must flow away from the motor with as little thermal impedance as possible. The existing Jetpack already has a liquid cooling system for the combustion engine. The cooling system can be reused for an electric motor. The cooling system should be effective as it is currently designed to expel large amounts of heat from the combustion engine.

3.5.1 Liquid Cooling

Liquid cooling allows a higher current density to be used [10], and the power density of the motor to be increased. To cool the stator and windings, an aluminium cooling jacket can be bonded to the surface of the stator. The cooling jacket has tunnels which run in a helix formation from one end of the jacket to the other [14]. A coolant is pumped through the jacket and then through a heat exchange system to remove the heat. The coolant is typically a water-glycol solution ranging from 30% to 50% glycol. The ethylene glycol acts as an anti-freeze for sub-zero temperatures. Table 3.2 shows a survey of liquid cooling used in electric vehicle traction motors. Inlet flow velocity has been calculated where possible. The flow velocity inside the motor depends on the internal geometry of the channels. Typical flow rates are from 5-10 L/min, and the typical maximum coolant temperature is 55°C.

Motor:	Nominal Power (kW)	Min. Coolant Flow Rate (L/min)	Max. Inlet Temperature (°C)	Inlet Area (mm^2)	Inlet Flow Velocity (m/s)
UQM PowerPhase 145	85	8	55	*no data*	*no data*
TM4 Motive MO120	37	6.7	55	44	2.5
Calmotors GP200WC	62	5	55	127	0.7
EVO-Electric AF-140	72	10	55	200	0.8
Siemens 1PV5135-4WS28	67	16	70	*no data*	*no data*

Table 3.2: Short survey of liquid cooling used in electric traction motors. Data from product datasheets [15-17].

It is useful to calculate an approximate limit of heat transfer using this type of liquid cooling. Once this limit is found, it can be used as a limiting parameter in motor simulations in order to maximise power density. The majority of the thermal impedance in liquid cooled motors occurs within the windings. If the windings are coated in insulation rated to 180°C, and an approximate temperature drop of 100 degrees between the centre of the windings and the stator yoke exists, then the stator yoke can be approximated to be 80°C. Good thermal contact is made between the aluminium jacket and the stator yoke. The heat transfer coefficient between two metallic surfaces with a 3 µm milled finish can be approximated as 0.11 $W/cm^2°C$ [10]. The use of thermal grease to fill in voids will be necessary to overcome the additional surface roughness due to the accuracy of lamination placement. If thermal grease is used with a 3 µm milled finish then the heat transfer coefficient can be doubled to 0.22 $W/cm^2°C$ [10]. However, since the lamination cut ends have unknown placement accuracy, the lower value of 0.11 $W/cm^2°C$ is more appropriate to use.

The heat transfer coefficient between the jacket and the coolant is estimated next. The average flow velocity is assumed to be 0.7 m/s through a rectangular bore within the aluminium jacket. The bore has a cross-sectional area of 128 mm^2 with a width of 16mm and a height of 8mm. The bore forms a helix running from one end to the motor to the other, and is packed tightly. The coolant properties need to be estimated first.

The properties of a 30% glycol solution at 55°C are [18]:

- Density: $\rho = 1018\, kg/m^3$
- Thermal conductivity: $k_f = 0.5010\, W/mK$
- Dynamic viscosity: $\mu = 9.311\times10^{-4}\ kg/ms$
- Prandtl Number: $\mathrm{Pr} = 7.100$

The hydraulic diameter D_H of the bore is

$$D_H = \frac{4A}{P} = \frac{4\times128\times10^{-6}}{48\times10^{-3}} = 10.67\ mm \qquad (3.3)$$

where A is the cross-sectional area of the bore and P is the wetted perimeter of the bore. Calculation of Reynolds number Re_D for an internal flow gives

$$\mathrm{Re}_D = \frac{\rho u_m D_H}{\mu} = \frac{1018\times0.7\times10.67\times10^{-3}}{9.311\times10^{-4}} = 8166 \qquad (3.4)$$

where u_m is the mean fluid velocity within the bore.

Since Re_D is much larger than the critical Reynolds number of 2300, the flow can be assumed to be turbulent [19]. This is desired as a turbulent flow increases the heat transfer coefficient for a given mean flow velocity. Next the friction factor f and the Nusselt number Nu_D can be calculated for a fully developed turbulent internal flow.

$$f = (0.79\ln(\mathrm{Re}_D) - 1.64)^{-2} = (0.79\ln(8166) - 1.64)^{-2} = 0.0333 \qquad (3.5)$$

$$Nu_D = \frac{(f/8)(\mathrm{Re}_D - 1000)\,\mathrm{Pr}}{1 + 12.7(f/8)^{0.5}(\mathrm{Pr}^{2/3} - 1)} \qquad (3.6)$$

$$Nu_D = \frac{(0.0333/8)(8166 - 1000)7.1}{1 + 12.7(0.0333/8)^{0.5}(7.1^{2/3} - 1)} = 66.03$$

The heat transfer coefficient from the bore surface into the coolant h_{bf} can be estimated as

$$h_{bf} = \frac{Nu_D \times k_f}{D_H} = \frac{66.03 \times 0.5010}{10.67 \times 10^{-3}} = 3100\, W/m^2K \qquad (3.7)$$

The actual surface area available for cooling is approximately half the surface area in the bore, which is also approximately equal to the surface area of the stator surface beneath the bore. This is a conservative estimate made since the heat flow is primarily from the inside surface of the water jacket. Taking this into consideration, the heat transfer coefficient per unit of surface area on the stator h_s can be approximated as

$$h_s \approx h_{bf} = 3100\, W/m^2K \qquad (3.8)$$

Next the thermal resistance from the stator to the jacket, and then from the jacket into the coolant fluid can be combined as thermal resistances for an equivalent area of 1cm^2.

$$R = 1/(3100 \times 10^{-4}) + 1/0.11 = 12.32\ K/W \qquad (3.9)$$

If the coolant is 55°C and the surface of the stator is 80°C, then the thermal resistance limits the heat transfer to

$$q_{\max} = \frac{\Delta T}{R} = \frac{25}{12.32} = 2.03\, W/cm^2 \qquad (3.10)$$

For less conservative estimates, the stator is assumed to be at 100°C (or alternatively the temperature of the cooling fluid can be made lower). Using the same analysis as above yields

$$q_{\max} = \frac{\Delta T}{R} = \frac{45}{12.32} = 3.65\, W/cm^2 \qquad (3.11)$$

This is a significant difference from the conservative estimate, and can occur due to the sensitivity of the heat transfer to small changes in stator temperature or the thermal impedance of the junction between the stator and the jacket.

It is worthwhile to survey the heat transfer limits of existing liquid cooled motors by analysing their dimensions, efficiency, and power curves in order to estimate the heat flow per unit of surface area on the stator. By comparing the theoretical results to real world motors, a more confident estimate of heat flow limitations can be made. Table 3.3 shows the results of a survey of high performance liquid cooled motors used in electric traction applications.

Motor:	**Nominal power (kW)**	**Nominal Efficiency (%)**	**Nominal Loss (kW)**	**Estimated Stator Area (cm^2)**	**Heat Flux (W/cm^2)**
UQM PowerPhase 145	85	93	6.4	1242	5.2 (2.8)
UQM PowerPhase 150	100	95	5.3	1908	2.8 (1.4)
UQM PowerPhase 200	115	93	8.7	1908	4.5 (2.3)
Calmotors GP200WC	62	96	2.6	1506	1.7
Calmotors GP150WC	45	96	1.9	1303	1.4
Symetron P-200	100	93	7.5	2796	2.7

Table 3.3: Survey of heat flow in liquid cooled motors. Bracketed values are estimations made since the efficiency was from the combined motor and driver. Data is from product datasheets [15, 16, 20].

These values are in a similar range to the theoretical estimates calculated above. The UQM motors efficiency maps included both the motor and driver together. It is difficult to determine how much of the losses were inside the motor, and how much were inside the driver. Because of this, a conservative estimation of the actual heat flux on the stator surface is made by assuming that half of the losses are in the motor driver. The results are shown in brackets in Table 3.3. In the Jetpack, there is already a liquid cooling system, as well as high speed airflow within the ducts that may assist with cooling. By considering the results from the theoretical calculations, and the results from the survey of motors, the heat flux used in modelling should be limited to less than 3 W/cm^2 for preliminary designs.

3.5.2 Other Cooling Optimisations

Apart from using liquid cooling, there are a number of other design features that can be used to increase the power density of the motor, or decrease the operating temperature. These are

- Encapsulation of windings: The windings are impregnated with a high thermal conductivity resin. This removes the air-gaps between windings, which is a major source of thermal impedance [10].
- Increased slot count: This decreases the cross-sectional area of each slot, and hence decreases the thermal impedance from the centre of the windings to the stator yoke [10].
- Through ventilation: The motor can also utilize the fan duct airflow. A number of holes/slots can be made in the stator to carry airflow from the fan ducts though the motor. Fins can be attached to the stator bore of outside-rotor motors.

Even though the airspeed within the ducted fans is very high (approximately 83 m/s), the use of liquid cooling still performs better than air cooling alone. By using a forced convection approximation from [10], and by approximating the length of the motor to be 20 cm, the heat transfer coefficient (in W/cm^{2}K) with air cooling can be calculated as

$$h_{air} = 3.89 \times 10^{-4} \sqrt{V / L} \qquad (3.12)$$

$$h_{air} = 3.89 \times 10^{-4} \sqrt{83 / 0.20} = 0.00792\, W / cm^2 K$$

where V is the air velocity in m/s and L is the axial frame length of the motor in metres. By using the same stator temperatures used in the previous section, and by setting the air temperature to 35°C, the heat transfer limit for air cooling is between approximately 0.36-0.52 W/cm^2 at the stator surface. Using fins will increase this; however it is clear that liquid cooling still has a significantly higher heat transfer limit.

Accurate calculation of heat transfer and fluid flow is more complex than the electromagnetic and mechanical design of motors [10]. For this reason it is difficult to perform quick and accurate calculations of heat flow. One of the main problems is the estimation of thermal resistance at the contact areas, such as the contact between the copper coils to the slot liner, and the slot liner to the stator yoke [10]. Consequently these thermal resistances become a major source of uncertainty. One method that can be used to obtain accurate results is to build a portion of the motor slots and perform empirical measurements of thermal impedances [10].

3.6 Survey of Existing Motors

A detailed survey of existing high performance motors was made to determine whether an off-the-shelf electric motor could be purchased and used to power the Electric Jetpack. Table 3.4 shows a summary of the survey containing the most important information relevant to this discussion. The full survey can be found in Appendix A.

Motor	Type	Nominal Power (kW)	Power Density (kW/kg)	Cooling Type	Speed at Nominal Power (RPM)
UQM PowerPhase 75	PMBL	45	1.10	Liquid	3000-7500
UQM PowerPhase 100	PMBL	100	1.10	Liquid	2000-4500
UQM PowerPhase 125	PMBL	45	1.10	Liquid	5000-8000
UQM PowerPhase 145	PMBL	85	1.70	Liquid	4200-8000
UQM PowerPhase 150	PMBL	100	1.10	Liquid	3700-4500
TM4 Motive MO120	PMBL	37	1.42	Liquid	6700-10000
Calmotors GP150WC	PMBL	43	0.98	Liquid	5300
Calmotors GP200WC	PMBL	59	1.18	Liquid	5200
Calmotors GP250WC	PMBL	96	1.75	Liquid	9200
Calmotors GP750WC	PMBL	151	1.06	Liquid	3000
EVO-Electric AF-140	PMBL	72	1.80	Liquid	3500
EVO-Electric AF-240	PMBL	145	1.73	Liquid	3500
AC Propulsion AC-150	Induction	50	1.00	Air	7000-8000
Enova EDU 90	Induction	30	0.46	Liquid	3600
LeTourneau SR 200	SRM	30	0.13	Air	?
Siemens 1PV5135-4WS28	Induction	67	0.74	Liquid	3000-8000
Siemens 1PV5138-4WS24	Induction	85	0.71	Liquid	3500-5500
BRUSA HSM6.17.12	Hybrid sync.	45	0.85	Liquid	4000
MES-DEA 200-330W	Induction	40	0.50	Liquid	?
Raser P-200	Induction	100	0.89	Liquid	5000-10000
Guang Dong M&C	PMBL	68	0.50	Liquid	?
DPMB M&C	PMBL	50	0.83	?	?
AIM-55 M&C	Induction	55	0.30	Air	4000
Enova EDM 240	Induction	150	0.52	Liquid	1800
Enova EDM 120	Induction	65	0.41	Liquid	1800
Yuneec Power Drive 40	PMBL	40	2.11	Air	2400
Yuneec Power Drive 60	PMBL	60	2.00	Air	2400
Strecker Motoren 990.8	PMBL	30	4.00	Air	?
AeroConversions	PMBL	37	1.63	Liquid	?
SkySpark	PMBL	60	1.50	Liquid	?
EM42	PMBL	38.5	1.32	Air	?

Table 3.4: Survey of motors used in electric vehicles and electric aircraft.

It can be seen that all bar one of the motors surveyed were below 2.5 kW/kg (the power density of the existing combustion engine). The exception is one motor which appeared to have a power density of 4 kW/kg; however contact with the manufacturer was unsuccessful.

The Yuneec PowerDrive 40 motor had a power density of 2.11 kW/kg. This is a specially designed motor used in all electric one and two-seater aircraft. Unfortunately, this motor does not operate at the speed required by the Jetpack. However, it is useful to take the power density figure for this motor and perform an analysis of motor weight. At 2.11 kW/kg the motor weight for the Electric Jetpack would be 71.1 kg. This is heavier than the current combustion engine, and it does not include the extra weight needed for a motor driver and any power cable weight. The motor driver for this motor obtains a power density of 5.7 kW/kg [21]. By extrapolation, the motor driver for the Electric Jetpack would need to weigh 26 kg. The motors from UQM Technologies in the survey had better power densities for their motor drivers. They are liquid cooled, and can achieve a continuous power density of 7.2 kW/kg [15]. Using this power density figure makes the controller weight only 20.8 kg. This brings the combined motor and driver weight to 91.9 kg. Including 5 kg for the power cable and 5 kg for miscellaneous parts makes the total weight 101.9 kg, or 36.9 kg heavier than the existing combustion engine and fuel tank. Such a tethered Electric Jetpack could still obtain flight, however it would only be able to carry a lighter pilot/load (<90 kg).

It is desirable to obtain a higher power density for the motor so that the pilot/load weight is not compromised. Since none of the motors surveyed could meet both the power density and speed requirements, the design of a custom motor was deemed worthwhile. This would allow optimisation of power density, and would allow the motor speed to be well matched.

3.7 Summary

The PMBLDC motor was shown to be the most suitable motor type for the Electric Jetpack due to it having the highest power density. This is confirmed by the fact that electric aircraft and many electric traction motors use the PMBLDC/AC motor type, as was shown in the motor survey. The PMBLDC motor has a 5-15% power density advantage over the PMBLAC motor. However, this power density advantage may be reduced as the electrical frequency of the motor increases (which is the result of higher motor speeds, or higher numbers of magnet poles). Liquid cooling can be used to design a high power density motor, and it was determined that the upper limit for heat flux removal when using an aluminium cooling jacket bonded to the stator is approximately 3 W/cm^2. A detailed survey of existing motors available for small electric aircraft and electric traction applications did not find any motors which met the power density and speed requirements for the Electric Jetpack. From this it was decided that the design of a custom motor was worthwhile. Given the best power densities from the motor survey, it is likely that the Electric Jetpack would only be feasible when powered from a tethered cable.

Chapter 4

Energy Storage Technology Investigation

4.1 Introduction

A high power density and high energy density energy storage technology is required for a non-cable bound Electric Jetpack to be feasible. The purpose of this investigation is to determine the status of current and upcoming energy storage technologies, and to comment on the suitability of applying different technologies on the Electric Jetpack. The energy storage technologies discussed are the battery, the fuel cell, and the ultra-capacitor. Simulations are used to determine the flight time that may be possible.

4.2 Battery Performance Parameters

It is useful to first document the various battery performance parameters and how these relate to the needs of the Electric Jetpack. More in-depth information about these parameters can be found in relevant texts [22]. The ideal battery will have a high power density, high energy density, flat discharge curve, long cycle life, and no self discharge. However, in reality there is always a trade-off between these parameters.

4.2.1 Power Density

The power density is the measure of how much power can be drawn from the battery per unit of mass. The power density of a battery is the most important parameter for the Electric Jetpack, due to the 150 kW of power needed to drive the ducted fans. For example, a NiMH battery with a power density of 500 W/kg would require the battery bank to weigh at least 300 kg. The power density requirement is what makes most current battery systems totally unsuitable for the Electric Jetpack.

4.2.2 Energy Density

The energy density is a measure of how much energy can be drawn from the battery per unit of mass or volume. The units for energy density can be stated in watt-hours per kilogram (Whr/kg) and watt-hours per litre (Whr/L). The available energy density of a battery decreases as the discharge rate of the battery increases. This is because increasing the discharge rate causes higher resistive and polarization losses in the battery [22]. This is important to consider since the Electric Jetpack will require a high discharge rate due to the high power density requirements. A higher energy density will allow greater flight times for the Electric Jetpack.

4.2.3 Discharge Profile

The discharge profile refers to the shape of the discharge curve (voltage vs. time). A flat discharge profile means that the output voltage of the battery will undergo very little change as the battery discharges. This is desirable for the Electric Jetpack as there will be tolerance limits for the bus voltage used by the motor driver so that rated current can be supplied to the motor.

4.2.4 Self Discharge

The self discharge of a battery refers to the rate at which a battery will lose its charge while not in use. Self discharge is usually approximated as a certain percentage per week, month, or year. The self discharge parameter is not a big concern for the Electric Jetpack, as the battery can be topped up with a recharging unit before flight. However, it is still desirable to have a relatively low self discharge rate.

4.2.5 Cycle Life

The cycle life of a battery refers to the number of times a battery can be discharged and charged until the available capacity of the battery reduces to an unacceptable level. Cycle life is usually stated for an 80% depth of discharge (DoD) [22]. Typically, a higher DoD results in a lower cycle life. A DoD of 100% could destroy or permanently damage many battery types. If the Electric Jetpack is used in applications where it is flown often, a high cycle life is important. If the Electric Jetpack is only flown occasionally, then the cycle life is not so important.

4.2.6 C-rate

Battery capacity is measured in amp-hours, and given the symbol “C”. The capacity is actually stated for a given “C-rate”, and the available capacity is dependent on the discharge rate. The C-rate of a battery is a measure of the rate at which a battery is discharged with respect to its capacity. For example a battery with a stated capacity of 100 Ahr, which is being discharged at a C-rate of 2, would have a discharge current of 200A. Since the capacity of a battery is dependent on the weight of active materials used, the C-rate is similar to power density.

4.3 Current Energy Storage Technology Status

A summary of currently available battery technologies is shown in Table 4.1. The data is from [22] and [7], and has been modified to reflect current performance figures. It can be seen that lithium-ion based technology holds the most promise for the Electric Jetpack due to its very high power density and energy density. The power density of all the other battery chemistries/systems is unsuitable for the Electric Jetpack. Ultra-capacitors are not technically batteries; however they share many similar characteristics, so they have been included in the same comparison. It can be seen that the power optimised ultra-capacitors have an extremely high power density. However, the low energy density means any flight time would be severely limited.

Energy Storage Technology	Energy Density (Whr/kg)	Energy Density (Whr/L)	Power Density (W/kg)	Discharge Profile @ C/5 rate	Cycle Life @ 80% DoD	Price (€/kWh)
Lead-acid						
Flooded	25 – 40	60 – 100	140 – 350	Flat	200 – 1500	100 – 190
VRLA	30 – 40	80 – 100	140 – 300	Flat	300 – 1000	100 – 190
Compressed	40 – 50	100	140 – 250	Flat	800 – 1500	35 – 50
Nickel-cadmium						
Power optimised	25 – 40	130	500	Very flat	800 – 1500	400 – 1000
Energy optimised	40 – 50	130	120 – 350	Very flat	800 – 1500	400 – 1000
Nickel-metal hydride						
Power optimised	40 – 55	80 – 200	500 – 1400	Very flat	500 – 2000	400 – 2000
Energy optimised	60 – 80	200 – 350	200 – 600	Very flat	500 – 2000	200 – 700
Nickel-zinc	60 – 80	200 – 300	500 – 1000	Flat	200 – 1000	500 – 800
Lithium-ion						
Power optimised	70 – 130	150 – 450	600 – 3000	Very flat	800 – 1500	700 – 2000
Energy optimised	110 – 220	150 – 450	200 – 600	Flat	800 – 1500	150 – 600
Lithium polymer	100 – 180	100	300 – 2000	Flat	300 – 1000	300 – 500
Ultra-capacitor						
Power optimised	3 – 5	3 – 10	2000 – 10000	Sloped	500k – 1M	1700 – 2300
Energy optimised	12 – 20	3 – 6	500 – 1000	Sloped	500k – 1M	1700 – 2300

Table 4.1: Comparison between currently available battery technologies [7, 22].

Each of the technologies can be better visualised using a Ragone chart. This type of chart visually shows the trade-off between power density and energy density. Figure 4.1 shows a Ragone chart made for the various energy storage technologies. The chart is based on data from [7] and [23], and has been modified to reflect current performance figures. The fuel cell has also been added to this chart. It is clear from the Ragone chart that lithium-ion based battery technology has an overall advantage in combining high power density and energy density. The fuel cell can provide a very high energy density, but it does not provide a very high power density.

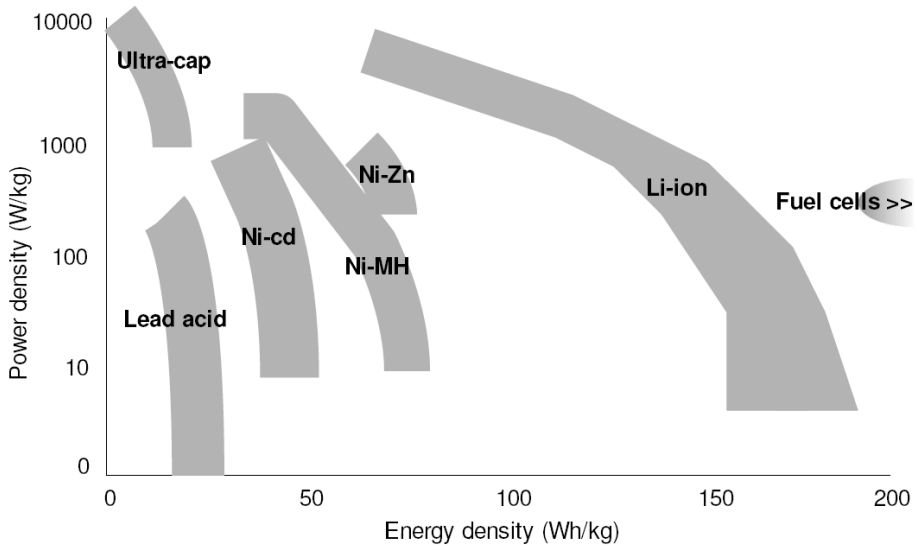


Figure 4.1: Ragone chart comparison between various energy storage technologies [7, 23].

4.4 Lithium-Ion Battery Technology Outlook

Lithium-ion based battery technology is currently the best available due to the combination of power and energy density. Table 4.2 shows a comparison between various lithium-ion based chemistries. Some of the chemistries are commercially available, while others are still under research and development. In terms of power density, the lithium iron nano-phosphate battery is best, with the cells from A123 Systems quoting >3 kW/kg power density [24].

Type	Companies	Status	Power	Energy	Safety	Cost
Lithium cobalt oxide	Various consumer applications	Limited automotive applications	Good	Good	Low, Moderate	High
Lithium nickel cobalt aluminium	JCI-Saft GAIA Matsuhita Toyota	Pilot	Good	Good	Moderate	Medium
Lithium iron phosphate	A123 Valence GAIA	Pilot	Excellent, Good	Moderate	Moderate	Medium
Lithium nickel cobalt manganese	Litcel Kokam NEC Lamillion	Pilot	Moderate	Moderate, Good	Moderate	Medium
Lithium manganese spinel	GS Yuasa Litcel NEC Lamillion EnerDel	Devel.	Moderate	Poor	Excellent, Good	Medium
Lithium titanate	Altairnano EnerDel	Devel.	Good	Poor	Good	High
Lithium manganese nickel titanate		Research	Good	Moderate	Excellent	Medium
Lithium manganese titanate	EnerDel	Research	Excellent	Excellent	Excellent	Medium

Table 4.2: Comparison between current and newly developed lithium-ion battery chemistries. Reproduced from [23].

The use of a vanadium-modified olivine structure in the lithium iron phosphate battery is an area of research with the potential to offer very high power densities. The rate capability of nano-structured $LiFeP_{0.95}V_{0.05}O_4$ is comparable to the best nano-sized $LiFePO_4$ formed solvo-thermally while using less than 1/3 the amount of carbon conductor [25]. What this means is that this technology could potentially provide another nano-structure based lithium iron phosphate battery which matches or improves upon what is currently available from A123 Systems.

The lithium titanate battery is another area where advances in power density are being made. One particular field of interest is in the development of the lithium manganese titanate battery. This battery combines a titanate anode with a manganese spinel cathode, and is currently being developed by EnerDel. According to EnerDel, their cells could provide more than a 50 C-rate continuous discharge [26]. Compare this with A123 Systems 30 C-rate continuous discharge [24], and the lithium

manganese titanate battery may become the most suitable battery for the Electric Jetpack in the near future.

Recently there has been a spur of research into using nano-wire materials to drastically increase the energy density of lithium-ion cells. For example, the use of silicon nano-wires as an anode material has shown an increase of ten times the energy density of traditional cobalt based lithium-ion cells [27]. However, this hasn't been followed through with a high discharge rate design which is required for the Electric Jetpack, and it is unclear whether this is possible at this time. A ten-fold increase in energy density could correspond to a flight time of approximately half an hour, assuming the power density requirement was satisfied. The calculation for this flight time approximation is

$$T_f = \frac{10E_{den}m}{p} = \frac{10 \times 220 \times 30}{150000} = 0.44\ hours \qquad (4.1)$$

where E_{den} is the best currently available energy density for cobalt based lithium-ion cells (in Whr/kg), m is the mass of the battery back (30 kg chosen to allow weight for the motors/driver), and p is the output power in watts. This is a very rough approximation, as the exact values to use for each of the variables cannot be determined, but it does provide a good idea of what future technology developments could offer for the Electric Jetpack.

4.5 Fuel Cell Technology Outlook

Fuel cells are an electrochemical cell, just like batteries; however they use an external fuel source, and react this with an oxidant which directly produces electricity [7]. This means that the external fuel source must be replenished once it is consumed. Currently there is a large volume of research being conducted in the development of hydrogen fuelled proton exchange membrane (PEM) fuel cells, so it is worth detailing the current and future trends of this technology. The driving force behind this research is to develop a clean renewable energy source for electric vehicles, and studies and reports have been made that assess their current and future potential (e.g. [7, 28]). Fuel cells already hold an advantage over existing battery

technologies since they can obtain higher energy densities [29]. There are many different types of fuel cells; these include proton exchange membrane (PEM) fuel cells, direct methanol fuel cells, alkaline fuel cells, phosphoric acid fuel cells, molten carbonate fuel cells, and solid oxide fuel cells. However the PEM fuel cell is the most suited to the Electric Jetpack because it can deliver higher power density, and offers the advantage of low weight and volume, when compared with other fuel cells [30]. It also holds good potential for future improvement due to the large amount of research and development being undertaken to make it a viable energy source for electric vehicles.

One of the major issues with PEM fuel cells is the hydrogen storage system. Hydrogen as a gas has a very low energy density, so some system of increasing the density of the stored hydrogen is needed to make a fuel cell system practical. The energy density that can be obtained depends on the hydrogen storage method used. A comparison of different possible storage systems is shown in Table 4.3. The U.S. Department of Energy goals are also listed in this table. While it seems these energy densities are all quite high when compared with current battery technology, it is important to note that this comparison does not include the fuel cell stack and associated components, and does not take into account the conversion efficiency of the fuel cell, which is approximately 50% when operating at peak power [28]. Currently, only the compressed hydrogen storage system is favoured to be used on board light vehicles, while liquid hydrogen has been demonstrated to be workable, it not considered to be widely accepted by automobile OEMs [28].

Hydrogen Storage System	Volumetric Energy Density (kWhr/L)	Gravimetric Energy Density (kWhr/kg)	Cost Per kWh (US$/kWhr)
DOE 2015 Target	2.7	3	2
DOE 2010 Target	1.5	2	4
Chemical Hydride	1.4	1.6	8
Complex Hydride	0.6	0.8	16
Liquid Hydrogen	1.6	2	6
Gaseous Hydrogen 10,000 psi	1.3	1.9	16
Gaseous Hydrogen 5,000 psi	0.8	2.1	12

Table 4.3: Approximate status of hydrogen storage technologies compared to DOE targets. Reproduced from [28].

When factoring in the fuel cell stack and other components needed to make the fuel cell system work, the available energy density decreases, but still remains higher than current lithium-ion battery technologies. Figure 4.2 and Figure 4.3 compares two different measurements of energy density for the PEM fuel cell system to the energy density of various battery chemistries. What becomes apparent is that the PEM fuel cell system only has an advantage in the gravimetric energy density. These comparisons show that the energy density of fuel cells definitely holds an advantage over currently available lithium-ion battery technology; however this may not be the case in the future. Considering the recent developments in nano-materials discussed in section 4.4, lithium-ion based technology could gain significant improvements in energy density and directly compete with the fuel cell systems.

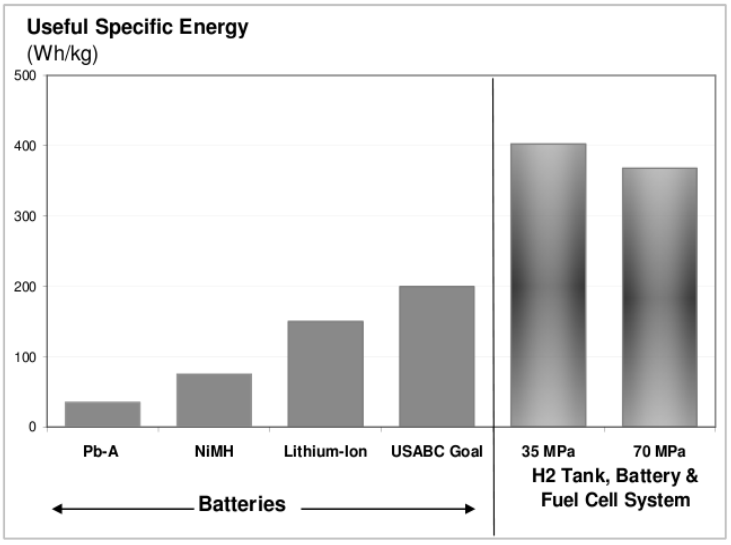


Figure 4.2: Specific energy comparison between the PEM fuel cell system and various battery chemistries. Reproduced from [29].

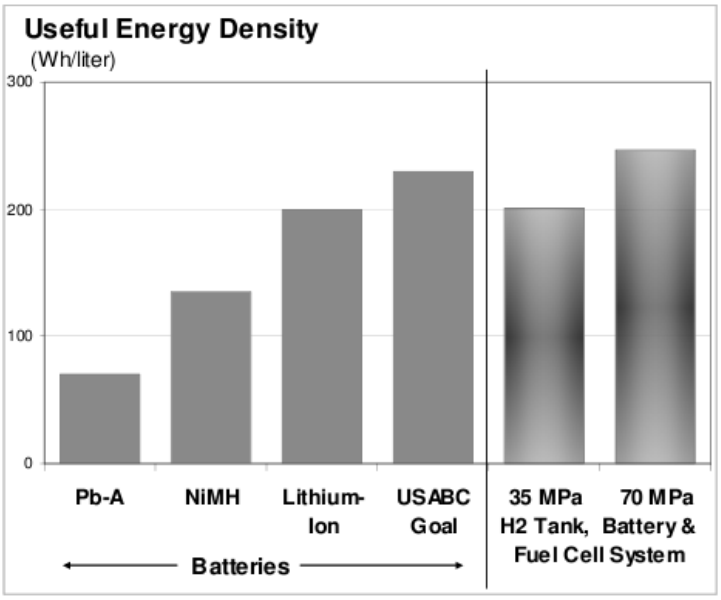


Figure 4.3: Energy density comparison between the PEM fuel cell system and various battery chemistries. Reproduced from [29].

The PEM fuel cell seems to be a viable system when observing its energy density. However, in its current state, it cannot supply the power density required for the Electric Jetpack. The currently available power density is approximately 500 W/kg, while the estimate for the year 2015 is 700–1100 W/kg [28]. Even at a power density of 1100 W/kg, the weight of the system for the Electric Jetpack would be 136

kg. For this reason, a PEM fuel cell system is unlikely to be a viable option for the Electric Jetpack for the foreseeable future.

4.6 Ultra-Capacitor Technology Outlook

Ultra-capacitors differ from conventional capacitors in that they replace the conventional dielectric with two layers of substrate with a separator in between these two layers. The dominant substrate material being used is micro-porous activated carbon [31]. Current ultra-capacitor technology seems unsuitable for the Electric Jetpack due to its low energy density. Future advances in ultra-capacitor designs will increase the available energy density and may make this a viable energy storage option for very short flights. To do this, the substrate double-layer material is replaced with new novel materials that have higher surface areas. A survey of ultra-capacitor technology approaches is shown in Table 4.4 below.

Technology Type	Material	Energy Density (Whr/kg)	Power Density (kW/kg)
Electric double-layer	Activated carbon	5-7	1-3
Electric double-layer	Carbon / organic electrolyte	6	5
Advanced carbon	Graphite carbon	8-12	1-2
Advanced carbon	Nanotube forest	unknown	unknown
Advanced carbon	Aerogel carbon	90	20
Pseudo capacitive	Metal Oxides	10-15	1-2
Hybrid	Carbon / metal oxide	10-15	1-2
Hybrid	Carbon / lead oxide	10-12	1-2

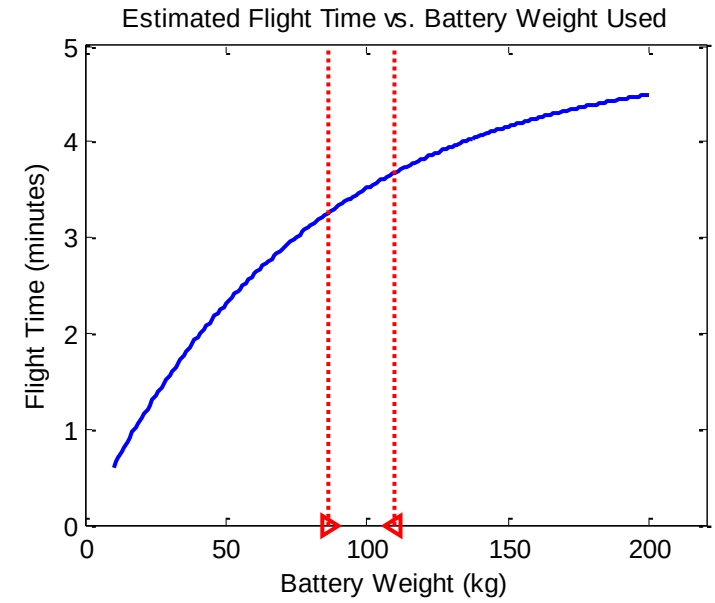
Table 4.4: Current ultra-capacitor research and design focus [31-33].

The carbon aerogel based ultra-capacitor holds some promise for use in the Electric Jetpack. This capacitor is speculated to be able to achieve 90 Whr/kg [32]. Currently, the only company producing aerogel based ultra-capacitors is Cooper Bussmann, and these capacitors only attain an energy density of 3 Whr/kg (calculated from manufacture's datasheet [34]).

4.7 Flight Time Simulations

In order for a particular energy storage technology to be viable in the Electric Jetpack, it must attain power densities in the thousands of watts per kilogram. In order to more accurately test the performance of various technologies, a flight time simulation script was written using MATLAB (version R2007b). The code is available in Appendix C. The simulations use the real world ducted fan thrust data, which was analysed in section 2.5. Only the lithium-ion battery and the ultra-capacitor offered power densities that warranted simulation. Other energy storage technologies fail by a large margin as their power density is too low. The first simulation uses the A123 Systems lithium iron nano-phosphate ANR26650M1 cell. Although the manufacturer states that the cell can obtain power densities greater than 3000 W/kg, the reality is that this does not refer to the continuously available power density. From analysing the discharge profile given in the manufacturer's datasheet, a continuous power density of approximately 2000 W/kg is a better estimate.

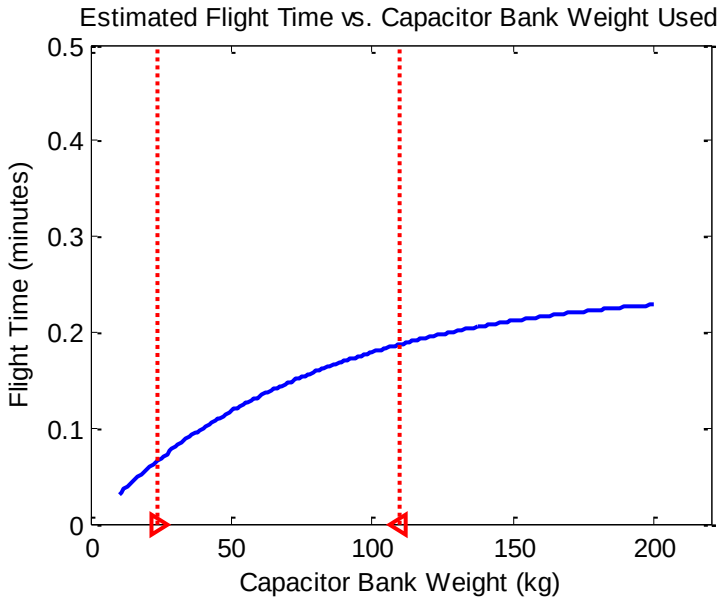
Figure 4.4 shows the results of a simulation for when the pilot/load weight is set to 85 kg. The two vertical red lines indicate the minimum and maximum battery weight restrictions. One restriction is formed from the power density requirements, while the other is formed from the maximum weight requirement (which is equivalent to a maximum fan speed of 7500 RPM). The results show that the battery can power the Electric Jetpack for between 3.2 minutes and 3.6 minutes of flight with a pilot/load of 85 kg. When the weight of the pilot/load was increased to 127 kg (which is the maximum for the combustion engine Jetpack), flight was not possible as the battery bank had to simultaneously weigh less than 68 kg and more than 127 kg to meet the weight and power density requirements. If the motor mass and drive mass were set to extremely light values (40 kg total), and the pilot was set to 85 kg, the flight time was still less than 5 minutes.



Simulation Setup:
- 90% system efficiency
- Pilot mass: 85 kg
- Frame mass: 35 kg
- Motor mass: 50 kg
- Drive mass: 26 kg
- Other mass: 15 kg
- Power density: 2 kW/kg
- Energy density: 108 Whr/kg

Figure 4.4: Flight time simulation using a lithium-ion battery.

The next simulation uses a prototype ultra-capacitor from APowerCap Technologies. The results are shown in Figure 4.5. This power source can theoretically obtain flight for 10 seconds; however in practice this is unpractical.



Simulation Setup:
- 90% system efficiency
- Pilot mass: 85 kg
- Frame mass: 35 kg
- Motor mass: 50 kg
- Drive mass: 26 kg
- Other mass: 15 kg
- Power density: 5 kW/kg
- Energy density: 5.5 Whr/kg

Figure 4.5: Flight time simulation using an ultra-capacitor bank.

It is important to note that these simulations show only what is required for level non-moving sustained flight, and there is no extra power capacity. The combustion engine based Jetpack uses a 150 kW engine, however the full power from this engine is not usually utilised. There is some reserve to account for different air densities, manoeuvrability, pilot weights, and future design changes. The simulations here do not include any reserve power capacity, and only show the absolute minimum requirements for sustained level flight under non-demanding operating conditions. For this reason, even the lithium iron nano-phosphate battery was not considered to be a feasible power source for the Electric Jetpack. Future improvements in power and energy density could change this outlook.

4.8 Summary

The Lithium iron nano-phosphate battery is currently the most suitable energy storage technology to power the Electric Jetpack. However, the flight time is only on the order of a few minutes, and this does not include any reserve power capacity which exists in the combustion engine based Jetpack. In order to improve the power density of the lithium-ion battery, a number of approaches are currently being researched. Once such approach is to combine a titanate anode with a manganese spinel cathode. Another is to use a vanadium-modified olivine structure in a lithium iron phosphate design. This research gives a good outlook for the improvement of power density in lithium-ion battery technologies. Advances are also being made in energy density. The use of silicon nano-wires as an anode material could potentially increase the energy density by a factor of ten. If this can be combined with a high power density design, then the lithium-ion battery could feasibly power the Electric Jetpack for approximately half an hour.

Ultra-capacitors currently do not have the energy density required to sustain flight in the Electric Jetpack beyond a few seconds. In reality these few seconds will be needed to just power up the Electric Jetpack, and will not contribute to flight time. The future outlook for ultra-capacitors indicates that higher energy densities will be developed. One possible method to accomplish this is to use carbon aerogel as the double-layer substrate. This may potentially increase the energy density to a maximum of 90 Whr/kg, however this would still only correlate to a flight time of a

few minutes. Because of this, it is unlikely that ultra-capacitor technology will be a feasible power source for the Electric Jetpack in the foreseeable future.

The hydrogen based PEM fuel cell currently does not have a high enough power density to allow flight. The power density target for the year 2015 is 700-1100 W/kg. But even taking this into account, it is still not high enough to power the Electric Jetpack. For this reason, the PEM fuel cell does not appear to be a feasible power source for the Electric Jetpack in the foreseeable future.

Chapter 5

Investigation into a Tethered Electric Jetpack

5.1 Introduction

It is clear that current electrical energy storage technologies do not provide the energy and power density required for the Electric Jetpack to be practical. Another option is to power the Electric Jetpack from a tethered cable. The purpose of this investigation is to determine the most suited currently available cable technology to use for the Electric Jetpack, and to research possible future cable technologies. The key considerations for the power cable are the cable mass, thickness, flexibility, durability, and price. The cable mass is important since the Electric Jetpack will need to support at least some of the cable weight as it flies. The cable thickness and flexibility are important since the cable will need to be able to flex and change direction as the Electric Jetpack moves. The cable durability is important since the cable will be repetitively flexing, and abrasion resistance will be required if any part of the cable is dragged on the ground. The price is unlikely to be an issue with currently available conventional cables, but could be a factor in alternative and future technologies such as super-conducting cable.

5.2 Bus Voltage and Current Requirements

The current and voltage required for the DC power supply bus will vary depending on the particular motor and motor driver used in the Electric Jetpack. A typical range of bus voltages is 120V to 650V, with 320V - 350V being very common for PMBL motors in the 50-100 kW range (based on motors investigated in the survey). This creates a complication for selecting the right power cable, as many voltage and current requirements now need to be considered.

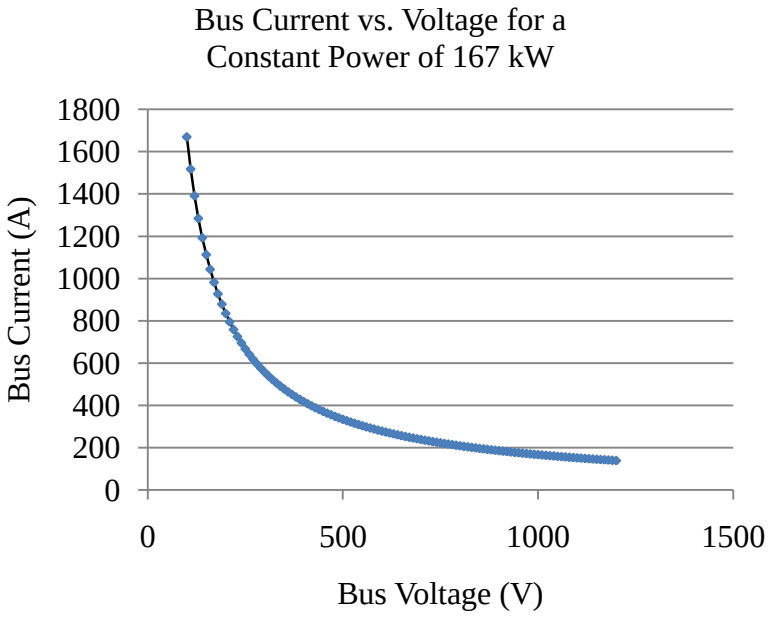


Figure 5.1: Bus current for a range of bus voltages.

Figure 5.1 shows a graph of the required maximum bus current for a range of bus voltages. The maximum mechanical power output is set at 150 kW. Assuming a 90% efficiency from the energy source to the motor output, the required maximum electrical input power is 167 kW. From observing Figure 5.1, it becomes clear that higher bus voltages should be favoured, particularly above 500V, as the current is proportional to the inverse of the voltage. Smaller currents will allow a thinner cable to be used, which in turn will allow weight savings and allow a more flexible cable. Using a bus voltage that is too high has some disadvantages. The first is that thicker insulation is needed in the motor windings and the power cable. Thicker insulation in the motor windings decreases the amount of bare copper that can be placed into each slot. Thicker insulation on the power cable may decrease the flexibility of the cable. Due to the nature of the relationship shown in Figure 5.1, there are diminishing returns when using higher voltages. For a very long cable, it may be better to include a step-down voltage converter on the Electric Jetpack to allow the voltage used in the cable to be in the thousands of volts range. A study into the expected weight of a possible step down converter located at the Jetpack end of the power cable to limit the motor voltage has not been made in this project. This is because the focus has been on other areas such as the motor design due to time constraints. However it could be considered if the Electric Jetpack project is further pursued.

5.3 Current Power Cable Technology Status

The following sections detail aspects of electric power cable design as it stands using today's technology, and how this applies to the requirements of the Electric Jetpack.

5.3.1 Resistivity

Figure 5.2 shows a resistivity comparison between various metal conductors. It becomes clear from the comparison as to why copper and aluminium are the two leading materials used for electrical wire. These two materials have a very low resistivity, and are both relatively cheap in comparison to silver and gold. They can also be drawn into wire, and resist corrosion.

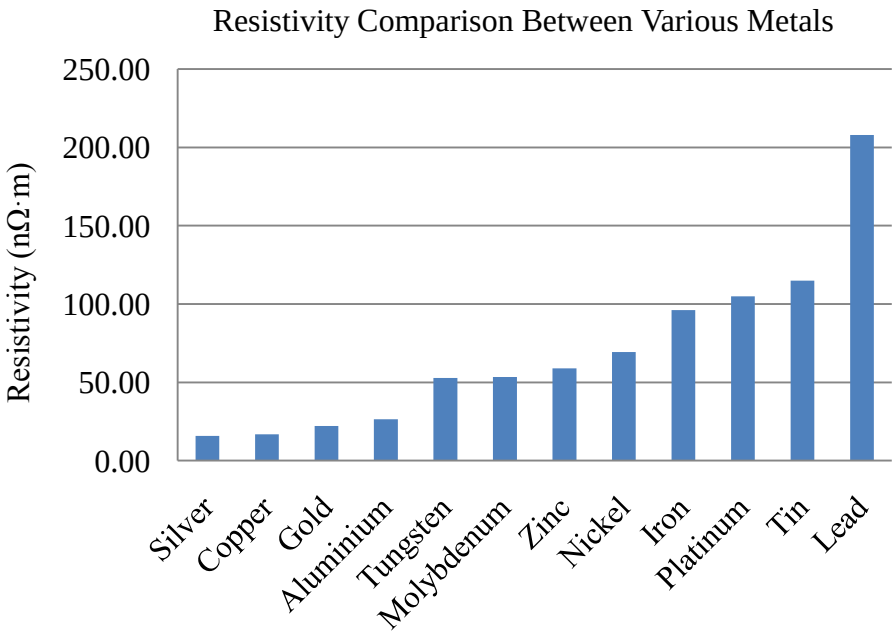


Figure 5.2: Resistivity comparison between various metal conductors. Data from [35].

5.3.2 Resistivity-Density Product

The resistivity of the material is not the only important factor. Clearly for the Electric Jetpack, the weight of the cable is very important. It is possible for a higher resistivity material to have a lower mass for the same resistance, due to its lower density. The disadvantage is that a thicker cable is needed. In order to quantify this, the resistivity-density product was used. Figure 5.3 shows the resistivity-density products for the same materials examined in Figure 5.2. A lower resistivity-density

product may be a better choice for the Electric Jetpack, providing that the cable can still be made flexible and durable.

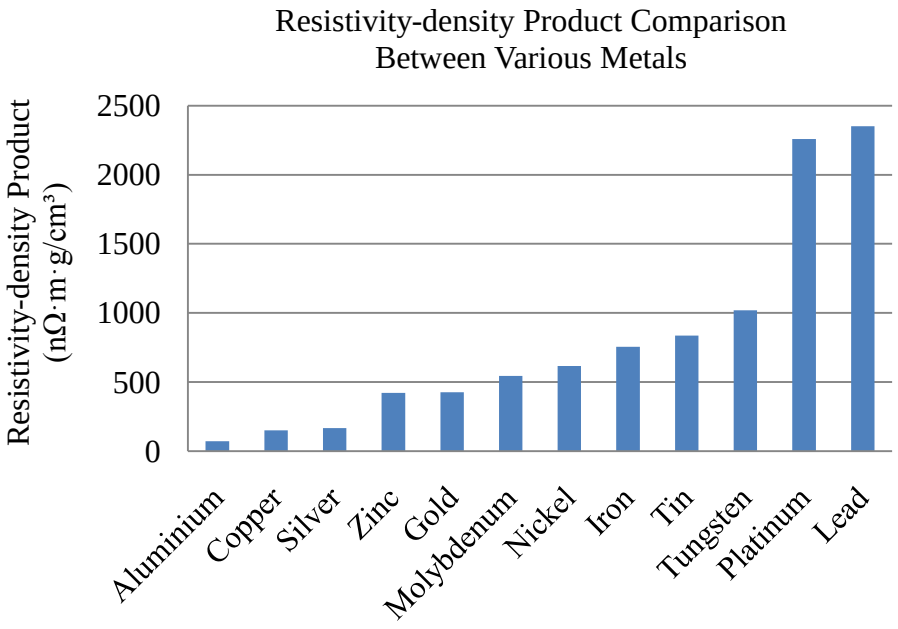


Figure 5.3: Resistivity-density product comparison between various metal conductors. Data from [35].

Copper, silver, and aluminium offer very low resistivity-density products. Out of these three metals, aluminium has the lowest resistivity-density product. In fact, bare aluminium wire only needs to weigh 48% of a bare copper wire in order to obtain the same DC resistance. However, some of this weight advantage is lost when the conductor is insulated because more insulation volume is required over the equivalent aluminium wire to cover the greater circumference [36].

5.3.3 Conductor Price

Aluminium and copper are both much cheaper than silver and gold. Between aluminium and copper, aluminium is cheaper. However, the type of cable construction and sheath will also have some effect on the cable price. As will be shown in the next section, aluminium is not a good option for the Electric Jetpack, and so the price between these two metals is not so important.

5.3.4 Cable Stranding and Flexibility

The flexibility of an electrical cable is largely dependent on the number of conductive strands used, and the type of conductor used. A greater the number of strands on the same cross-sectional area produces a more flexible cable with a smaller minimum bend radius. There are a number of advantages to using a flexible cable. Firstly the cable does not affect the manoeuvrability of the Electric Jetpack. Secondly, much of the cable mass can be supported on the ground, with only the cable directly below the Electric Jetpack needing to be lifted. Thirdly, a more flexible cable will last longer and be more durable as it can be flexed without fracturing of the conductor material. When selecting a flexible cable, it is not necessary to state the number of strands desired. The cables are categorized according to their minimum bend radius. For the Electric Jetpack cable, a minimum bend radius of < 30 cm should work fine.

Aluminium is a good choice for lowering the cable mass, but aluminium strands have a low bending life compared to copper strands [36]. This makes aluminium a poor choice for flexible cables that are required to withstand repetitive flexing without fracturing. Many electrical cable manufacture's supply special cables designed for high flexibility. Product data can be found on manufacturer's websites (e.g. [37, 38]). These cables are designed with a large number of conductive strands and a softer insulation compared to standard cables. The conductive strands are made from copper or a copper alloy. Special copper alloys such as beryllium copper offer highly elastic/spring-like properties that allow the strands of copper to remain unchanged after many flexing cycles [39]. Regular annealed copper flexible cables also perform well, and should be sufficient for the Electric Jetpack.

A sufficient level of abrasion resistance on the cable sheath will be required since rubbing may occur between the cable and the floor or other objects. This presents a safety hazard if the cable sheath is damaged and high voltages are exposed. There is a trade off between the abrasion resistance, flexibility, and current rating. A more flexible cable requires that a softer material is used in the insulation, which decreases the abrasion resistance of the cable. Alternatively, less insulation can be used but this again reduces the abrasion resistance. If more insulation material is used, then

the ability to dissipate heat is reduced, and the current rating reduced. The cable insulation should be chosen according to the expected use of the cable with accordance to manufacturer's recommendations.

5.3.5 Current Rating

The current rating of a cable is the maximum amount of continuous electrical current a cable can safely carry without suffering any short term or long term damage. The current rating of an electrical cable varies depending on the conductor, sheath, and insulation material used, as well as the physical layout of the cable (e.g. open air or underground installation). The current rating is limited by the maximum temperature the cable sheath or insulation is allowed to rise to. According to the joint Australia and New Zealand standard AS/NZS 3808, "Insulation and sheathing materials for electric cables" , this temperature is set at 90 degrees Celsius for most modern materials such as polyvinyl chloride (PVC) and polyethylene, with the exception of where both the cable insulation and sheath is made from PVC the temperature limit is 75 degrees Celsius.

It is not practical to give current ratings for every type of flexible cable here since there is much variation between manufacturers and models. Instead it is better to give a specific example in order to get an approximate idea of the size of cable that will be required. For this example, the cable chosen is the "Hyperflex" cable from General Cable New Zealand. This cable is made from a stranded copper conductor, ethylene-propylene rubber insulation, and vinyl-nitrile rubber sheath [38].

Conductor Cross-Sectional Area (mm^2)	Current Rating (A)	Cable Diameter (mm)	Cable Mass (kg/m)	Minimum Bend Radius (mm)
16	109	10.4	.25	65
25	149	12.4	.37	75
35	182	13.7	.49	85
50	215	15.9	.66	100
70	275	17.6	.94	110
95	319	20.1	1.17	125
120	374	21.9	1.41	135
150	429	24.3	1.85	150
185	484	27.0	2.25	165
240	561	30.2	2.89	185
300	660	33.3	3.59	200

Table 5.1: Current ratings for "Hyperflex" cable [38].

Table 5.1 shows the current ratings for this cable as provided by the manufacturer. The current ratings are based on the joint Australia and New Zealand standard AS/NZS 3008. This table can be used in conjunction with the bus current requirements in Figure 5.1 in order to estimate the cable size for a given bus current and voltage. For example, if the bus voltage of the motor driver is 320 V, then the corresponding bus current is 520 A. Using the table gives a cable cross-sectional area of 240 mm^2, a cable diameter of 30.2 mm, and a minimum bend radius of 185 mm. If the Electric Jetpack has to support a five metre length of power cable (keeping in mind that two cables are required for DC current), then the corresponding cable weight is 28.9 kg. The same analysis done at a bus voltage of 1000 V results in a cable weight of only 4.9 kg. This clearly shows the advantage of using a higher bus voltage.

5.4 Power Cable Technology Outlook

5.4.1 High Temperature Superconducting Wire

High temperature superconducting (HTS) wire allows significantly greater current densities compared to conventional copper wire. Second generation (2G) wire can conduct 150 times the electrical current of copper wires of the same size [40]. However, HTS wire when used as a power cable needs to be mounted in special designs that include liquid nitrogen cooling in order to maintain the superconductivity of the wire. There are currently a number of demonstrations of

this technology being used, such as the 609 meter, 138 kV, high-temperature superconductor power cable that was energized on April 22, 2008 at Holbrook substation, New York. However, these demonstrations focus on high voltage high power cables for electrical transmission grids. The Electric Jetpack requires a much lower power cable in comparison to these examples. With future improvements in superconducting wire technology, it could become possible to power the Electric Jetpack from a superconducting cable. The cooling system could be set on the power supply side of the cable and not contribute to the weight of the Electric Jetpack. In order for this to be worthwhile, the cable would need to be flexible and lightweight, which may be a challenge as the cryostat is required to surround the cable.

5.4.2 Nano-Structured Cable

The use of carbon nanotube structures as a power cable is a future technology with the ability to provide a lightweight cable with a high tensile strength. The molecular structure of carbon nanotubes greatly affects their conductivity. The conductivity of carbon nanotubes can either be semiconducting or metallic in nature. The problem lies in the reliable creation of the metallic type, without the semiconducting types. A project at Rice University is being carried out in an effort to create a quantum power cable which could theoretically conduct electricity up to 10 times better than traditional copper wire and weigh one-sixth as much [41]. Under an agreement with NASA, Rice's Carbon Nanotechnology Laboratory is to attempt to produce a 1-meter-long prototype of quantum wire [41].

5.5 Applications

Although this project focuses on the engineering aspects of determining the feasibility of an Electric Jetpack, it is useful to mention some possible applications for a tethered Electric Jetpack. One potential application is thrill rides. The Electric Jetpack could be operated continuously as there is no fuel required. The Electric Jetpack would also be quieter than the combustion engine version, so it may appeal to more people for this kind of application. Another potential application is in rescue operations on multi-storied buildings. Rescuers could obtain access to roof tops, or the Electric Jetpack could be controlled remotely so that the person needing to be rescued could strap themselves to the Electric Jetpack and be flown to safety.

5.6 Summary

The weight of a 5 meter power cable using a 1000 V supply bus voltage is only 4.9 kg. Future technologies such as high temperature superconducting wires and nano-structured cables offer some promise to improve upon the weight of a copper cable. The nano-structured wire does not require a cryostat as it operated at room temperature, so it may offer a better and more practical power cable solution. Current flexible copper cables provide a feasible solution to powering the Electric Jetpack, especially if a high bus voltage is used. Potential applications for the tethered Electric Jetpack include thrill rides and rescue operations.

Chapter 6

Development of a PMBLDC Motor Simulation Program

6.1 Introduction

No current off-the-shelf motors could be found which satisfied the approximate power density and speed requirements for the Electric Jetpack. It was decided to pursue a custom motor design instead. The first step in this process was to develop a simulation program that would facilitate the optimisation of a PMBLDC motor for the Electric Jetpack. By doing this, the feasibility of an Electric Jetpack could be clarified, and the first step in the design process could be made. This chapter details the theory and processes used to create the PMBLDC simulation program. It also covers some issues discovered relating to the motor time constant and current commutations. The simulation program was written using MATLAB (version R2007b), and can calculate many important motor parameters, such as the back-EMF waveform, inductance, resistance, resistive loss, core loss, torque output, efficiency, etc.

6.2 Theory

Methods and calculations used by the PMBLDC simulation program have been based on theory from relevant texts [8-11]. More in-depth theory and operation of PMBLDC motors can be found in those texts. The PMBLDC motor simulation program requires that some physical dimensions of the motor are defined by the user, as well as the materials used (such as the type of magnet material and laminations). Using this input from the user, the program uses the following process to calculate the performance of a particular motor design. The source code is supplied in Appendix E.

6.2.1 Air-gap and Magnet Flux Density

The first step is the calculation of the remanence flux density from the magnets. All units for the following equations are SI units unless stated otherwise. The remanence flux density is

$$B_R = B_{Rroom}(1+\alpha_t(T_m - T_{room})) \tag{6.1}$$

where B_{Rroom} is the remanence flux density at room temperature, α_t is the temperature coefficient, T_m is the operating temperature of the magnets, and T_{room} is room temperature. Next the air-gap flux density is calculated as

$$B_g = B_R \frac{K_l \frac{A_{mp}}{A_{gp}}}{1+\mu_R K_l \frac{A_{mp}}{A_{gp}} \frac{g'}{L_m}} \tag{6.2}$$

where K_l is the flux leakage factor, A_{mp} is the magnet area per pole, A_{gp} is the air-gap area per pole, μ_R is the relative permeability of the magnet material, g' is the effective air-gap length, and L_m is the thickness of the magnets. The area per magnet pole is calculated at a radius of one third the thickness of the magnet. The effective air-gap length is approximately equal to the air-gap length in surface mounted magnet motors [10]. The flux leakage factor is approximated to be 0.90 as per recommendation in [9]. This equates to 10% flux leakage. The flux density inside the magnets is calculated as

$$B_m = \frac{B_g A_{gp}}{K_l A_{mp}} \tag{6.3}$$

6.2.2 Yoke Width Approximations

The flux that crosses the air-gap into the stator yoke per pole, and the flux the flows into the rotor yoke per pole can be calculated as

$$\phi_g = A_{gp} B_g \tag{6.4}$$

$$\phi_{ry} = A_{mp} B_m \tag{6.5}$$

The required width for the stator yoke is approximated as

$$W_{sy} = \frac{\phi_g}{2 L_{st} B_{stator} K_{st}} \tag{6.6}$$

where L_{st} is the axial stack length of the stator, B_{stator} is the maximum desired flux density in the stator yoke, and K_{st} is the stacking factor. The stacking factor is a measure of how much of the cross-sectional area of the laminations is filled with the core material (insulation and packing produces gaps between the laminations). The width of the rotor yoke is approximated as

$$W_{ry} = \frac{\phi_{ry}}{2 L_{st} B_{rotor}} \tag{6.7}$$

where B_{rotor} is the maximum desired flux density in the rotor yoke. In surface mounted magnet motors, the rotor yoke does not undergo a large variation in flux density as the rotor rotates, as the permanent magnets are fixed into position on the rotor yoke, and the gap between the stator and rotor is large. Because of this, the use of laminations in the rotor yoke may not be necessary, and instead a solid steel yoke could be used. According to recommendation from [10], the flux density for a solid carbon steel rotor yoke should be limited to 1.5 T.

6.2.3 Tooth and Stator Yoke Flux Waveforms

Determining the required tooth width is a more difficult task. It was found that very simple approximations provided in literature did not produce accurate results due to the effects of slotting at the tooth tips. In order to more accurately determine the tooth width required, a more complex algorithm was developed based upon the "tooth flux accumulation" method described in [10], and the approximation of air-gap flux density due to slotting in [9]. The tooth flux is found by numerical integration of the air-gap flux density. This integration is repeated at each angle between the rotor and stator. The air-gap flux density is modified at each rotation position to account for the effects of slotting. Figure 6.1 shows an example of the tooth flux waveform calculated over one electric period for a 14 pole, 12 slot motor.

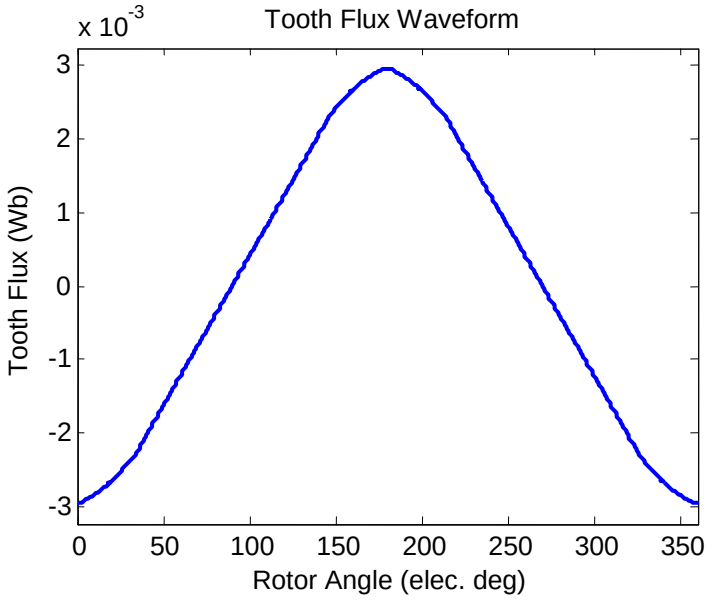


Figure 6.1: Tooth flux waveform example.

The maximum value on the tooth flux waveform is then used to accurately size the width of the tooth. The calculation is

$$W_t = \frac{\phi_{t\max}}{L_{st} B_{stator} K_{st}} \tag{6.8}$$

where $\phi_{t\max}$ is the maximum tooth flux. The tooth flux waveform is also used to determine the stator yoke flux waveform. The process used is to rotate the tooth flux

waveform for each tooth by the electrical angle between each tooth, and solve for the yoke flux waveform by setting the sum of the flux entering and leaving the stator yoke to zero. The solution for the stator yoke flux waveform becomes

$$\phi_{sy}(\theta) = \frac{1}{N_s}\sum_{k=1}^{N_s}(N_s - k)\phi_{t(k+1)}(\theta) \tag{6.9}$$

where N_s is the number of slots (or teeth) and $\phi_{t(k+1)}$ corresponds to the tooth flux waveform on the $k^{th}+1$ tooth. Figure 6.2 shows an example of the stator yoke flux waveform calculated using the tooth flux waveform in Figure 6.1.

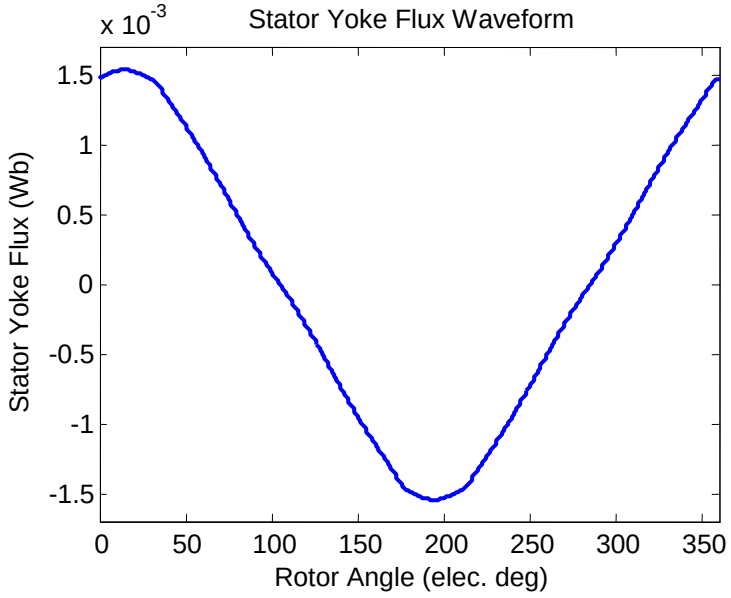


Figure 6.2: Stator yoke flux waveform example.

The width of the stator yoke can then be recalculated using the stator yoke flux waveform. The calculation is

$$W_{sy} = \frac{\phi_{sy\max}}{L_{st}B_{stator}K_{st}} \tag{6.10}$$

where $\phi_{sy\max}$ is the maximum stator yoke flux from the waveform.

6.2.4 Phase Back-EMF Waveform

The single tooth back-EMF is the back-EMF that would be measured if a single turn was wound around a single tooth and the rotor rotates at rated speed. The single tooth back-EMF can be used to calculate the phase back-EMF of an arbitrary distribution of coils. The method is described [10] . The single tooth back-EMF waveform is calculated using

$$e_{tooth} = \frac{d\phi_t}{dt} \tag{6.11}$$

where ϕ_t is the tooth flux waveform calculated previously. The distribution of the windings is defined by the user, and stored into a coil table. The coil table contains a list of each coil, including each coil's start slot, finish slot, and number of turns. A negative number of turns indicate that the coil is wound in the opposite direction.

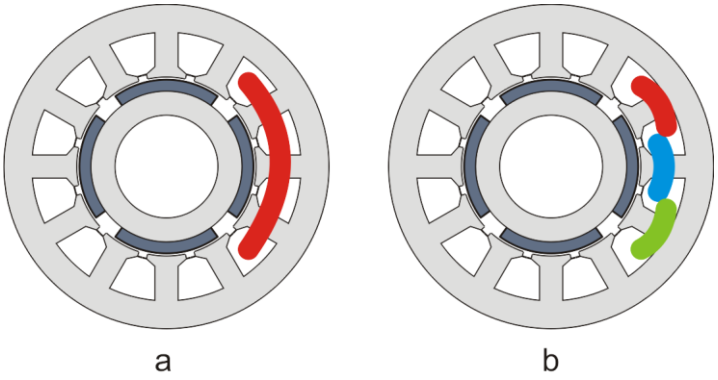


Figure 6.3: Decomposition of a single coil (a), into three single tooth equivalent coils (b).

The first step is to decompose the phase winding into a set of single tooth equivalent coils. A graphic representation of decomposing a coil is shown in Figure 6.3. As an example, if a coil starts in slot three, ends in slot six, and has 10 turns, then it can be decomposed into three single tooth equivalent coils, each having 10 turns. The first coil goes from slot three to four, the second coil goes from slots four to five, and the last coil goes from slots five to six. The back-EMFs for the conductors in slots four and five cancel each other out as they point in opposite directions. In this way the three coils are equivalent to the single coil from slots three to six, and will experience

the same total back-EMF when connected in series. The total phase back-EMF is then found from the superposition of the back-EMF for each single tooth equivalent coil that belongs to the phase winding. The back-EMF for each single tooth equivalent coil is found by the appropriate rotation and scaling of the tooth back-EMF from Equation 6.11. Table 6.1 shows an example of a coil table for a distributed LRK phase winding in a 14 pole, 12 slot motor. The term LRK is from the names of the designers Christian Lucas, Ludwig Retzbach, and Emil Kuerfuss, and is a popular split phase sector winding used in BLDC motors for RC model aircraft [42, 43]. It is popular because it minimises the copper in the end-turns and is simple to wind. Figure 6.4 shows the corresponding phase back-EMF obtained using the above methods. The actual back-EMF waveform would be smoother looking when compared to this parametric approximation, as the fringing of magnetic flux would ensure there are no instantaneous jumps in the back-EMF waveform. However, for quick parametric simulation purposes, the approximate waveform is used.

Coil Number	Slot Start	Slot Finish	Number of Turns
1	1	2	10
2	2	3	-10
3	7	8	-10
4	8	9	10

Table 6.1: Coil table example for a distributed LRK phase winding.

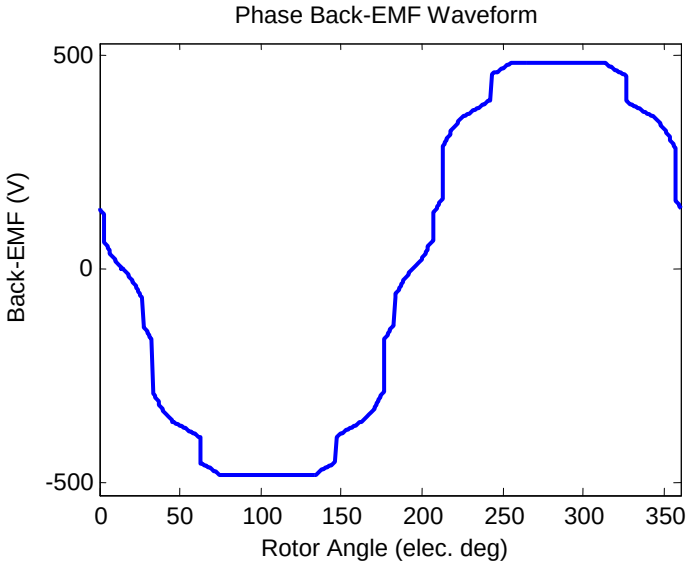


Figure 6.4: Phase back-EMF example for a distributed LRK winding.

6.2.5 Tooth and Slot Area

The tooth cross-sectional area is needed in order to calculate the area per slot that is available for windings. It is also required to calculate hysteresis and eddy current losses within the teeth. The method used to find the tooth area is by integration of areas formed by curves which are defined by the geometry of the motor. The MATLAB code for this process is included in Appendix E. Figure 6.5 shows the areas computed for both inside and outside-rotor designs. The areas within the triangles are only used when computing the core losses within the teeth. These areas are not subtracted from the area of the stator yoke when calculating the stator yoke loss. They represent an additional component of core loss at the transition between the teeth and the stator yoke. An detailed description can be found in [10].

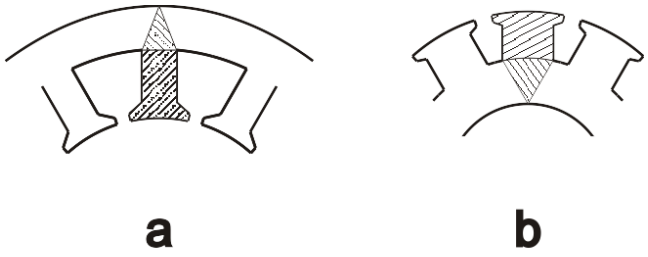


Figure 6.5: Tooth area and augmented tooth area for (a) inside-rotor, and (b) outside-rotor designs.

6.2.6 Phase Current

The slot fill-factor and RMS current density is defined by the user. The slot fill-factor measures the proportion of each slot cross-section which is filled with copper. Typical values for slot fill-factor are from 40% to 50 % [9]. The copper area per winding conductor in a three-phase machine is calculated as

$$A_{wire} = \frac{A_{slot} N_s K_{ff}}{3N_c} \tag{6.12}$$

where A_{slot} is the area per slot, N_s is the number of slots, K_{ff} is the fill-factor, and N_c is the number of conductors per phase.

If all conductors are connected in series in each phase, then the RMS phase current is calculated as

$$I_{ph} = A_{wire} J \tag{6.13}$$

where J is the desired RMS current density. From the RMS phase current, the phase current waveform is determined, and depends on whether the windings are connected in star or delta configuration. Figure 6.6 shows an example of the calculated phase current waveforms for star and delta connected windings.

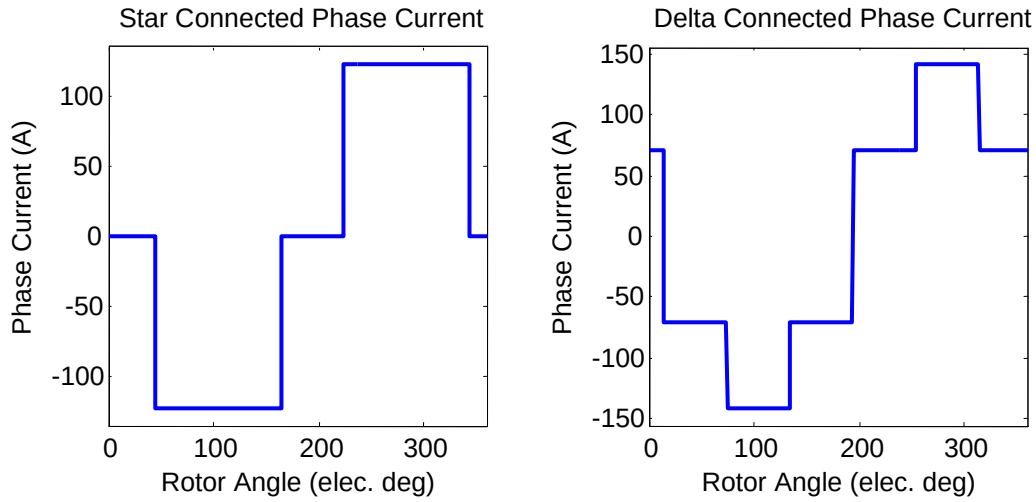


Figure 6.6: Star connected and delta connected phase current waveforms.

6.2.7 Torque

The torque is calculated using the back-EMF and current waveforms. Knowing the speed of the motor and applying conservation of energy results in a torque waveform of

$$\tau_{ph}(\theta) = \frac{i_{ph}(\theta) e_{ph}(\theta)}{\omega} \tag{6.14}$$

where $i_{ph}(\theta)$ is the instantaneous phase current, $e_{ph}(\theta)$ is the instantaneous phase back-EMF, and ω is the rotational velocity (assumed constant). The total torque waveform can then be calculated by summing each phase torque waveform together. Figure 6.7 shows an example of a total torque waveform generated using the above

process for a 14 pole, 12 slot, PMBLDC motor with distributed LRK windings. The torque ripple is the consequence of the interaction between the slot openings, the magnet arc size, the winding distribution, and the current waveform. Finite element analysis combined with a six-step drive current simulation would give a more accurate result for the torque ripple since the parametric model does not take into account the fringing of the magnetic flux, the reluctance of the stator yoke, or the commutation imperfections in the motor current waveform.

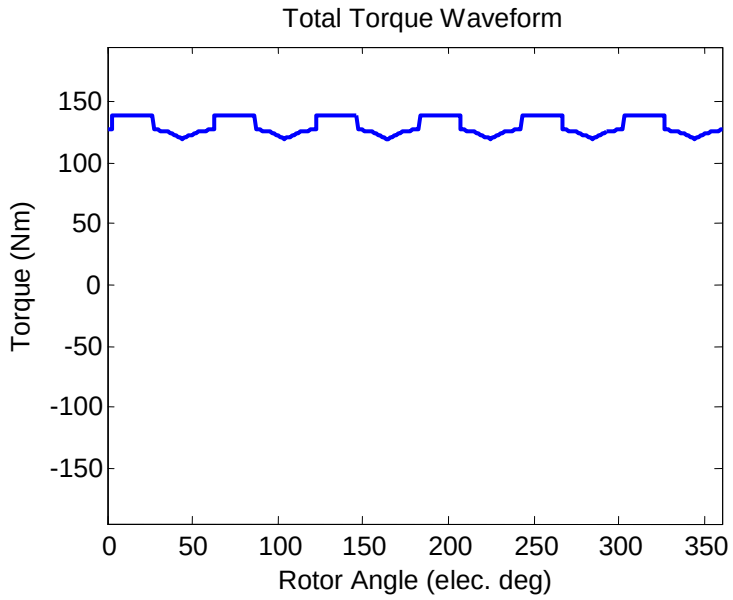


Figure 6.7: Total torque waveform example.

6.2.8 Phase Resistance

Phase resistance is calculated by summing the resistance contribution from each coil defined in the coil table. The first step is to calculate the length of copper in each coil. The coil span angle is the angle formed between the two slots which hold the coil ends. The coil span angle can be calculated as

$$\alpha_{coil} = \frac{2\pi}{N_s} s \tag{6.15}$$

where s is the number of slots that the particular coil spans. The windings are approximated to be concentrated in the centre of each slot, and the end turns are approximated using a modified circular arc from the slot openings, as per

recommendation in [10]. An example is shown in Figure 6.8 with a coil span of three slots.

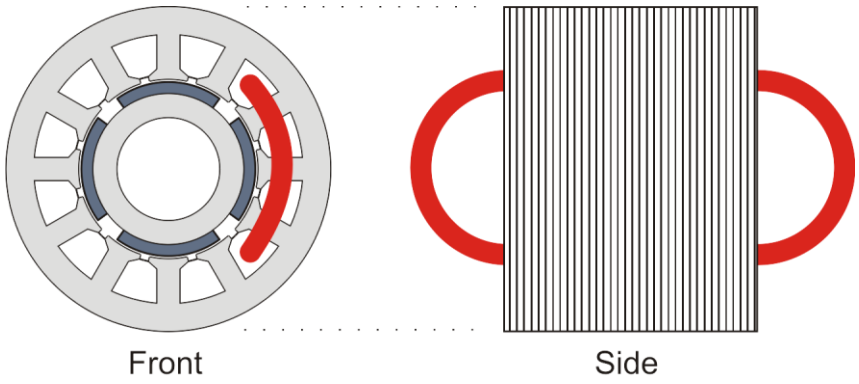


Figure 6.8: Geometry approximations for coils.

The length of each end turn is

$$l_{end} = \frac{\pi \alpha_{coil} r_{wind}}{2} \tag{6.16}$$

where r_{wind} is the average radius of the windings. The total length of copper per turn is

$$l_{turn} = 2l_{end} + 2L_{st} \tag{6.17}$$

The length of each coil is calculated as the number of turns per coil multiplied by the length per turn. The total length of copper winding per phase is the sum of the lengths of each coil belonging to the phase, which is defined in the coil table. The phase resistance at room temperature is

$$R_{rt} = \frac{\rho_{cu} l_{ph}}{A_{wire}} \tag{6.18}$$

where ρ_{cu} is the resistivity of copper at room temperature, l_{ph} is the length of the winding per phase, and A_{wire} is the cross-sectional area of the winding. Motor windings operate at much higher temperatures than room temperature, so the

resistance must be adjusted to reflect the increase in resistivity of copper at the operating temperature. The phase resistance at operating temperature is

$$R_{ph} = R_{rt}(1+\alpha_{cu}(T_{wind} - T_{room})) \tag{6.19}$$

where α_{cu} is the temperature coefficient for copper, T_{wind} is the temperature of the windings, and T_{room} is room temperature. The total resistive power loss for the three phase motor is

$$P_{loss-R} = 3R_{ph}I_{ph}^2 \tag{6.20}$$

6.2.9 Motor Weight

The motor weight is needed to determine the power density of the motor. The mass of the windings is

$$M_{copp} = 3l_{ph}A_{wire}D_{copp} \tag{6.21}$$

where D_{copp} is the density of copper. The mass of the teeth is

$$M_{teeth} = A_{tooth}L_{st}D_{yoke}K_{st}N_s \tag{6.22}$$

where A_{tooth} is the cross-sectional area of each tooth and D_{yoke} is the density of the stator lamination material. The mass of the stator yoke for the inside-rotor motor is

$$M_{sy} = D_{yoke}K_{st}L_{st}\left(\pi R_{so}^{\ 2} - \pi(R_{so} - W_{sy})^2\right) \tag{6.23}$$

where R_{so} is the outside stator radius. The mass of the rotor yoke for the inside-rotor motor is

$$M_{ry} = D_{stl}L_{st}\left(\pi(R_{ri} + W_{ry})^2 - \pi R_{ri}^{\ 2}\right) \tag{6.24}$$

where R_{ri} is the inside-rotor radius and D_{stl} is the density of carbon steel (assuming that a solid steel rotor is used). The mass of the magnets for the inside-rotor motor is

$$M_{mag} = k_{ang} D_{mag} L_{st} \left(\pi R_{ro}^2 - \pi (R_{ro} - L_m)^2 \right) \tag{6.25}$$

where k_{ang} is the angular magnet arc fraction, D_{mag} is the density of the permanent magnet material, and R_{ro} is the rotor outside radius. Very similar equations are used to calculate the mass of the stator yoke, rotor yoke, and magnets for the outside-rotor motor, and can be found in the code in Appendix E. The total motor mass is

$$M_{total} = M_{copp} + M_{teeth} + M_{sy} + M_{ry} + M_{mag} \tag{6.26}$$

The total mass calculated here does not include the weight of the bearings, cooling components, and any other structural supports or miscellaneous parts.

6.2.10 Core Losses

The core loss is calculated using a modified version of the Steinmetz equation to account for non-sinusoidal flux density waveforms [9]. The equation for core loss is

$$P_{core} = C_h f B_p^{n+mB_p} + \frac{C_e}{2\pi^2} \left\langle \left(\frac{dB}{dt} \right)^2 \right\rangle \tag{6.27}$$

where C_h is a material dependant constant for the hysteresis loss, C_e is a material dependant constant for the eddy current loss, f is the frequency of the flux density waveform, B_p is the peak flux density, n is a material dependant constant, m is a material dependant constant, and the last portion of the equation is the mean square value of the derivative of flux density with respect to time. The core loss equation has units of W/kg (assuming the material dependant constants were calculated from core loss curves with the same units), and has to be multiplied by the mass of the stator yoke or the teeth in order to determine the total core loss for a particular part of the motor. The stator yoke and teeth undergo different flux density waveforms as

shown in section 6.2.3, so the core loss for each must be calculated separately before adding the result together to obtain the total core loss. The material dependant constants were determined from graphs of core loss which are provided by the manufacturer. The core types defined in the PMBLDC simulation program are

- Silicon-iron standard: M19 0.014 inch
- Silicon-iron thin gauge: Arnon5 Special 0.005 inch
- Cobalt-iron thin gauge: Hyperco 50A 0.006 inch

The first is a general purpose lamination suitable for frequencies up to 400 Hz. Above 400 Hz it is recommended to use thin gauge silicon-iron laminations [44]. The thinner laminations decrease eddy current losses (which increase proportional to the frequency squared). The disadvantages of using thin gauge laminations are that the stacking factor decreases and the cost of the laminations increase [44]. Cobalt-iron laminations have a higher saturation flux density (approximately 2.0 T compared to 1.5 T for standard silicon-iron). This allows the widths of the teeth and stator yoke to be decreased, and hence weight savings can be made. Cobalt-iron laminations are very expensive compared to silicon-iron laminations, so they are usually used only if they are absolutely essential [45].

6.2.11 Inductance

The self inductance of each phase can be split up into three separate components: the air-gap inductance, the end-turns inductance, and the slot leakage inductance. The air-gap inductance is formed due to the component of flux that crosses the air-gap due to the windings acting alone. This flux which crosses the air-gap due to the stator windings is also called the armature reaction flux. The air-gap inductance is calculated using the method set out in [9]. This method can be used with different winding distributions, and makes use of the coil table that has been defined by the user. The phase winding is first decomposed into equivalent single tooth coils as described previously in section 6.2.4. The number of turns required for each tooth is stored into an array. Each term in the array is divided by the smallest number of turns out of all the single tooth coils, so that the number of turns for each tooth is normalised to the smallest number of turns per coil. An example of the array s is

shown in Table 6.2. This example is for a 4 pole, 12 slot motor with concentrated full-pitch windings that span three slots. There are 10 turns on each coil, so when it is normalised to the lowest number of turns per coil, the value for each tooth in the s array is one. Negative values occur when the coil is wound in the opposite direction.

Tooth j	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
$\mathbf{S_j}$	1	1	1	-1	-1	-1	1	1	1	-1	-1	-1

Table 6.2: Normalised single tooth equivalent coil array.

By using the s array, the air-gap inductance can be calculated as

$$L_g = \frac{N^2}{R_{pt}} \sum_{j=1}^{N_s} sign(s_j) \left[s_j - \frac{1}{N_s} \sum_{n=1}^{N_s} s_n \right] \tag{6.28}$$

where, N is the base number of turns used to normalise the s array and R_{pt} is the reluctance between each tooth and the rotor yoke. The reluctance between each tooth and the rotor yoke is approximated as

$$R_{pt} = \frac{g}{\mu_0 A_{gt}} + \frac{L_m}{\mu_R \mu_0 A_{mt}} \tag{6.29}$$

where A_{gt} is the area of the air-gap per tooth, A_{mt} is the area of the magnet pole per tooth, μ_0 is the permeability of free space, and μ_R is the relative permeability of the permanent magnet material.

The slot leakage inductance is due to flux leaking across slots and slot openings, instead of crossing the air-gap. It is found by first determining the number of conductors present in each slot for a given phase. This is found from the coil table, and stored into the array z. An example of the array for a 4 pole, 12 slot motor with concentrated full-pitch windings is shown in Table 6.3. There are 10 turns per coil, which makes 20 conductors per slot, as there are two coil sides in each slot.

Slot j	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**11**	**12**
$\mathbf{Z_j}$	20	0	0	20	0	0	20	0	0	20	0	0

Table 6.3: Conductors per slot array.

The next step is the calculation of the slot permeance coefficient, which depends on the distribution of the conductors within the slot [10]. An approximation for the slot permeance coefficient for a uniform distribution of conductors is

$$p_{slot} = \frac{d}{3w} + \frac{d_{tt}}{w_{to}} \tag{6.30}$$

where d is the slot depth, w is the slot width, d_{tt} is the tooth-tip depth, and w_{to} is the tooth-tip opening width. A graphical representation of these dimensions is shown in Figure 6.9.

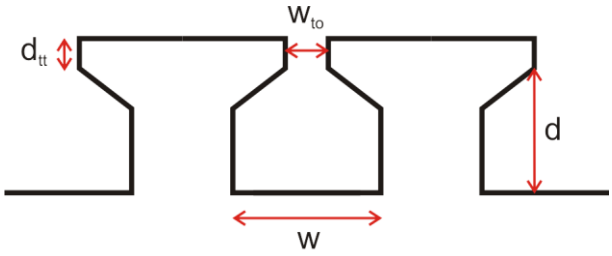


Figure 6.9: Geometry used to calculate the slot permeance coefficient.

The slot leakage inductance per phase can be calculated as

$$L_{sl} = \sum_{j=1}^{N_s} \mu_0 L_{st} p_{slot} z_j^2 \tag{6.31}$$

The end-turns inductance is due to the flux which circulates the end-turns which stick out of each end of the motor stack, as shown in Figure 6.8. The end-turns inductance is approximated by summing up the end-turns inductance for each coil listed in the coil table for a particular phase. The end-turns inductance is calculated as

$$L_{et} = \sum_{k=1}^{N_c} \frac{\mu_0 n_k^2 \alpha_k r_{wind}}{2} \ln\left(\frac{4\alpha_k r_{wind}}{GMD} - 2\right) \tag{6.32}$$

where N_c is the number of coils, n_k is the number of turns on the k^{th} coil, α_k is the coil span angle for the k^{th} coil, and *GMD* is the geometric mean distance between the conductors and the coil cross-section. The *GMD* can be approximated using a square cross-section [10]. The *GMD* is

$$GMD = 0.447\sqrt{A} \tag{6.33}$$

where A is the cross-sectional area of the winding. Finally the total self inductance per phase can be calculated as the sum of the three inductance components.

$$L_{ph} = L_g + L_{sl} + L_{et} \tag{6.34}$$

The mutual inductance is due to flux from one phase winding linking into another phase winding. The following calculation solves for the magneto-motive-force (MMF) at the rotor yoke due to a 1 ampere excitation of a phase winding with respect to the stator yoke. The MMF is

$$MMF = \frac{-N}{N_s} \sum_{k=1}^{N_s} s_k \tag{6.35}$$

where N is the base number of turns per coil, N_s is the number of slots, and s_k is the normalised single tooth equivalent coil array as described in section 6.2.11. The resulting tooth flux for the k^{th} tooth is

$$\phi(k) = \frac{MMF + s_k N}{R_{pt}} \tag{6.36}$$

The normalised single tooth equivalent coils for the first phase winding are rotated and stored in the new array n. The mutual flux linkage and the mutual inductance

are equal since a 1 ampere excitation was used. The mutual flux linkage and the inductance is

$$\phi_m = M = \sum_{k=1}^{N_s} \phi(k) n_k N \tag{6.37}$$

Using the above algorithm to determine the mutual inductance between two phases in a three-phase, concentrated full-pitch winding arrangement yields that the mutual inductance is -1/3 of the air-gap inductance, which is what is expected for this winding type [10].

6.2.12 Line-line Parameters

The line-line inductance and resistance are required to determine the time constant of the motor windings, and is useful when simulating motor currents. The line-line inductance for the star connected three-phase PMBLDC motor is

$$L_{LL} = 2L_{ph} - 2M \tag{6.38}$$

The line-line resistance is

$$R_{LL} = 2R_{ph} \tag{6.39}$$

6.2.13 Demagnetisation

Demagnetisation can occur when the armature reaction flux opposes the permanent magnet flux so much that the operating point of the magnet is pushed into the “knee” region. Figure 6.10 shows the magnetic characteristics of N3575 neodymium magnet material. The knee region can be seen when operating the magnet at very high temperatures. If the operating point of the magnet falls into this region then it will recoil out of this position along a lower operating curve.

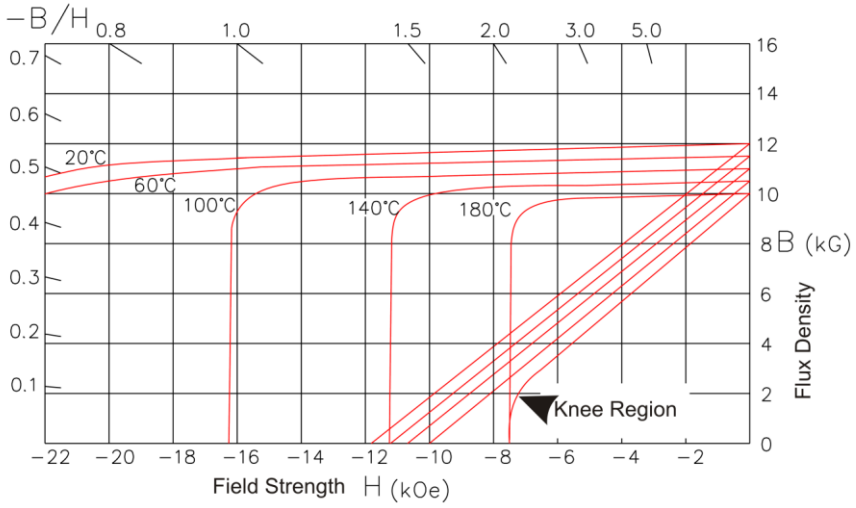


Figure 6.10: Neodymium N3575 magnetic characteristics [46].

In order to prevent demagnetisation the operating point of the magnet under maximum armature reaction must be calculated to check that it does not lie within the knee region. The tooth flux array for the 1 ampere excitation from Equation 6.36 is multiplied by the maximum phase current so that the maximum armature reaction tooth flux per phase in the k^{th} tooth is

$$\phi_{tw-ph}(k) = \phi(k) I_{\max} \tag{6.40}$$

where $I_{\max}$ is the peak phase current. Assuming a star connected winding, the total armature reaction tooth flux for the k^{th} tooth is the sum of two phases combined (reluctance of the yoke is ignored), and is equal to

$$\phi_{tw}(k) = \phi_{tw-ph}(k) + \phi_{tw-ph}(k + n_{ph}) \tag{6.41}$$

where n_{ph} is the number of slots between phases. The maximum armature reaction tooth flux out of all the teeth $\phi_{tw\max}$ is found from the array, and used to determine whether demagnetization occurs when the permanent magnet is aligned beneath the tooth in opposition to the armature reaction tooth flux. The flux in the tooth due to the magnet acting alone $\phi_{t\max}$ is already known from the tooth flux analysis presented in section 6.2.3.

The flux flowing out of the magnet directly under the tooth due to the magnet acting alone is conservatively approximated as

$$\phi_{mag} = \frac{\phi_{t\max}}{K_l} \tag{6.42}$$

The operating point of the magnet can then be calculated using superposition as the magnet equivalent circuit is linear (the reluctance and non-linearity of the yoke material is ignored for this approximation). The operating point of the magnet is

$$H_{mag} = \frac{(\phi_{mag} - \phi_{tw\max}) - B_R A_{mt}}{\mu_0 \mu_R A_{mt}} \tag{6.43}$$

If $H_{mag} < H_{knee}$ then the operating point of the magnet has shifted into the non-linear knee region, and demagnetization will occur. This demagnetisation prediction is only an approximation. Accurate demagnetisation prediction can be achieved using FEM software to account for the non-linearity and reluctance of the stator and rotor yokes and the fringing of flux.

6.2.14 Performance Figures

The power density is calculated as

$$P_{den} = \frac{\tau_{avg}\omega}{M_{total}} \tag{6.44}$$

where τ_{avg} is the average torque, ω is the rotational velocity, and M_{total} is the total motor mass. The efficiency is calculated as

$$Eff = \frac{\tau_{avg}\omega}{\tau_{avg}\omega + P_{loss-R} + P_{loss-C}} \tag{6.45}$$

where P_{loss-R} is the total resistive power loss and P_{loss-C} is the total core loss. The effects of friction are not included. Also of interest is the heat flux that must be removed from the stator surface.

The heat flux is

$$\dot{q} = \frac{P_{loss-R} + P_{loss-C}}{A_{stator}} \tag{6.46}$$

where A_{stator} is the surface area of the stator. The electrical time constant for the motor is

$$\tau = \frac{L_{LL}}{R_{LL}} \tag{6.47}$$

6.3 FEM Integration

The PMBLDC motor design simulation program was programmed to store the calculated geometry of the motor into an AutoCAD file. The AutoCAD geometry can be imported into FEM analysis software in order to verify the parametric model, and to perform any fine tuning of the motor design. The geometry can also be displayed to the user at the time of simulation so that any errors in geometry can be easily found.

6.4 Time Constant Issues

The PMBLDC motor has a disadvantage when it comes to handling the motor inductance. If the time constant for the motor is too high, then the phase currents can be limited, and may not reach the peak value required to sustain rated torque. It is unclear what constitutes a motor with a time constant that is too high. This is because the motor currents not only depend on the time constant, but also the motor driver voltage. The higher the motor driver voltage, the faster the currents rise within the motor windings, and the allowable time constant can be increased. In order to investigate the effects of the motor time constant in PMBLDC motors, a motor current simulation program was written using MATLAB (version R2007b). The code is included in Appendix D.

6.4.1 State Space Modelling

The motor is connected in star configuration as shown in Figure 3.2. The coupled circuit equations are

$$\begin{bmatrix} v_{as} \\ v_{bs} \\ v_{cs} \end{bmatrix} = \begin{bmatrix} R_{ph} & 0 & 0 \\ 0 & R_{ph} & 0 \\ 0 & 0 & R_{ph} \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + p \begin{bmatrix} L_{aa} & L_{ab} & L_{ac} \\ L_{ba} & L_{bb} & L_{bc} \\ L_{ca} & L_{cb} & L_{cc} \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix} \qquad (6.48)$$

where

v_{as}, v_{bs}, v_{cs} are the voltages over each phase,

R_{ph} is the resistance per phase,

i_a, i_b, i_c is the current in each phase,

p is the time derivative operator,

L_{xy} is the inductance (self or mutual) between phases x and y,

e_a, e_b, e_c is the back-EMF in each phase.

Due to the symmetry between the windings, and the fact that surface mounted PMBLDC motors have non-salient rotors (the rotor reluctance does not change with rotor angle), the mutual inductances are equal, and the self inductances are equal [8]. This allows the coupled circuit equations to be simplified to

$$\begin{bmatrix} v_{as} \\ v_{bs} \\ v_{cs} \end{bmatrix} = \begin{bmatrix} R_{ph} & 0 & 0 \\ 0 & R_{ph} & 0 \\ 0 & 0 & R_{ph} \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + p \begin{bmatrix} L-M & 0 & 0 \\ 0 & L-M & 0 \\ 0 & 0 & L-M \end{bmatrix} \begin{bmatrix} i_a \\ i_b \\ i_c \end{bmatrix} + \begin{bmatrix} e_a \\ e_b \\ e_c \end{bmatrix} \qquad (6.49)$$

where L is the self inductance of each phase and M is the mutual inductance between two phases. The state space equations need to be solved for each commutation period in the six-step drive. Each commutation period has slightly different differential equations to be solved since the voltages applied to each phase changes, and one phase free-wheels in the diode. For the first commutation period, the voltage applied to phase A is +Vs/2, the voltage applied to phase B is –Vs/2, and phase C is left to free-wheel until the current reaches zero. During free-wheeling

phase C is connected to the negative supply voltage. By solving Equation 6.49 for these conditions, the differential equations for the currents during the first commutation period is

$$p\begin{bmatrix} i_a \\ i_b \end{bmatrix} = \frac{-R_s}{L-M}\begin{bmatrix} i_a \\ i_b \end{bmatrix} - \frac{1}{3(L-M)}\begin{bmatrix} -2 & 1 \\ 1 & 1 \end{bmatrix}\begin{bmatrix} (v_{ab} - e_{ab}) \\ (v_{cb} - e_{cb}) \end{bmatrix} \qquad (6.50)$$

The current in phase C is simply found by applying Kirchhoff''s current law to the centre node of the star connected phases. The equations can be solved using the Runge-Kutta method (or any other appropriate method). The other five commutation periods have the same form as the above equation, only the subscripted phase letters change. The full set of solutions for the six-step drive connected to a PMBLDC motor can be found in Appendix B, and is based on the solutions from relevant texts [8, 10].

6.4.2 Time Constant Effects

The state-space equations for the motor currents were programmed into the motor current simulation program, and a number of simulations were made to investigate the relationship between the motor time constant, supply voltage, motor current, and torque. Figure 6.11 shows the ideal current shape that is obtained when there is no time constant. Figure 6.12 shows the phase current when the time constant is only 1% of the electrical period. This fraction of the electrical period was called the "time constant fraction" for this report. A small time constant fraction may occur during low speed operation (low electrical frequencies). The supply voltage was calculated so that the steady state current would be 8 A without any PWM chopping. It can be seen that the phase current follows the ideal motor current waveform shape as shown in Figure 6.11 very closely. The ripple in the current is due to commutation of current by the six-step drive. The current ripple causes a ripple in the output torque, which is called the commutation torque ripple.

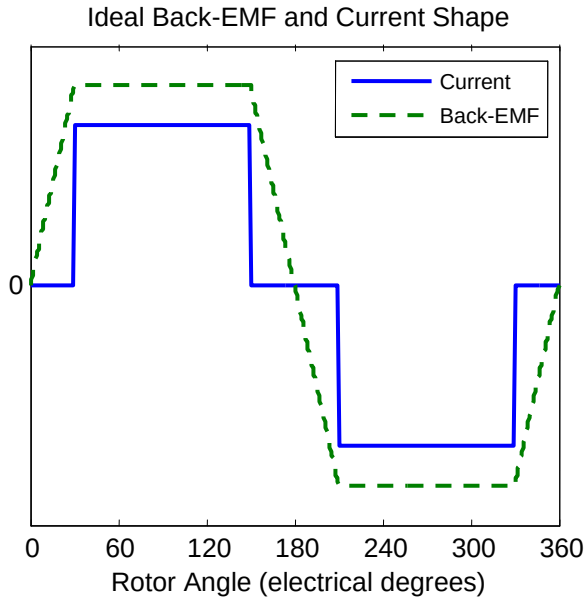


Figure 6.11: Ideal phase back-EMF and current shapes for the PMBLDC motor.

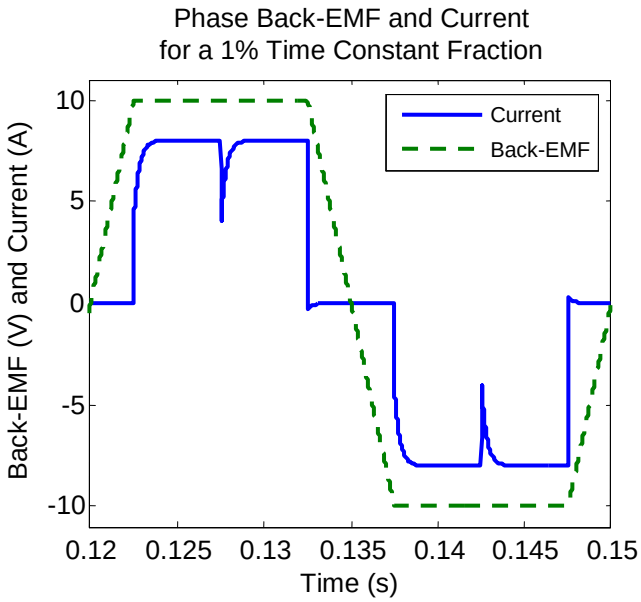


Figure 6.12: Phase back-EMF and current simulation with a small time constant fraction.

If the time constant fraction for a motor is too large, or the motor is operated at speeds that are too high, then the current does not have time to reach the rated current required to produce rated torque. Figure 6.13 shows the simulation results when the time constant fraction is set to 100%.

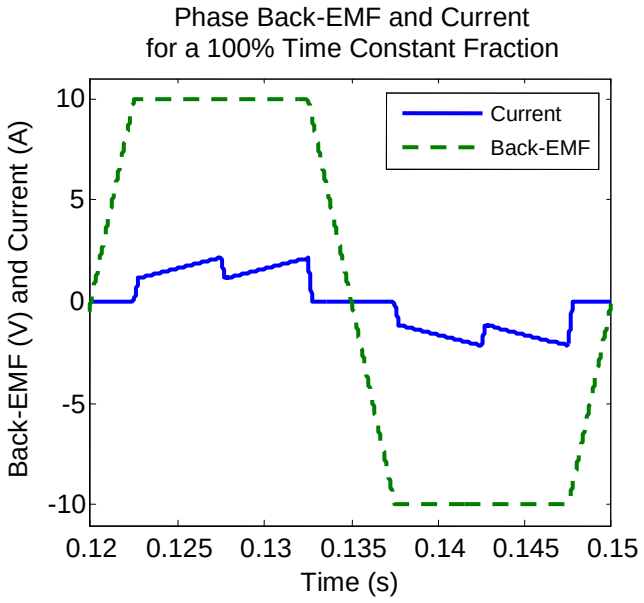


Figure 6.13: Phase back-EMF and current simulation with a larger time constant fraction.

It can be seen that the current does not reach the steady-state value of 8 A as it did in the previous simulation. The consequence of this is that the output torque of the motor only reached a quarter of the rated torque. From this it might be concluded that a time constant fraction of 100% is too large for a practical motor. However, this is not true, and will be shown by example.

There are two methods that can be used to restore rated current to the PMBLDC motor. The first is to increase the supply voltage of the motor, and the second is to use phase advance techniques. Phase advance works by energising the phase windings before the back-EMF has reached its peak value. This allows the equivalent voltage over the motor inductance to be much higher, and hence the current rises more rapidly. Figure 6.14 shows the results of adding a 20 degree phase advance to the motor simulation. It can be seen that just over half the rated current can be supplied to the motor, and a small phase shift is introduced between the phase current and the back-EMF. This example shows that phase advance is not entirely effective, and increasing the supply voltage is required to restore rated torque to the motor in this case.

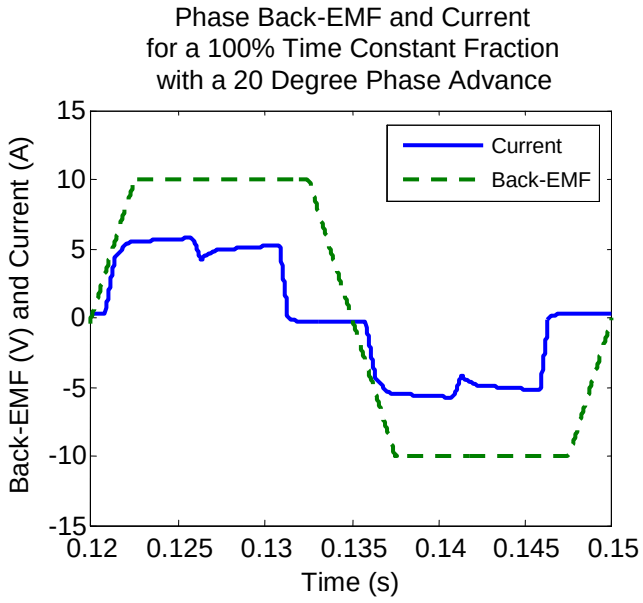


Figure 6.14: Current improvement when using phase advance.

By increasing the supply voltage by a small margin, the rated current can be restored. Figure 6.15 shows the simulation result when increasing the supply voltage by 15%. It can be seen that rated current is restored in this case.

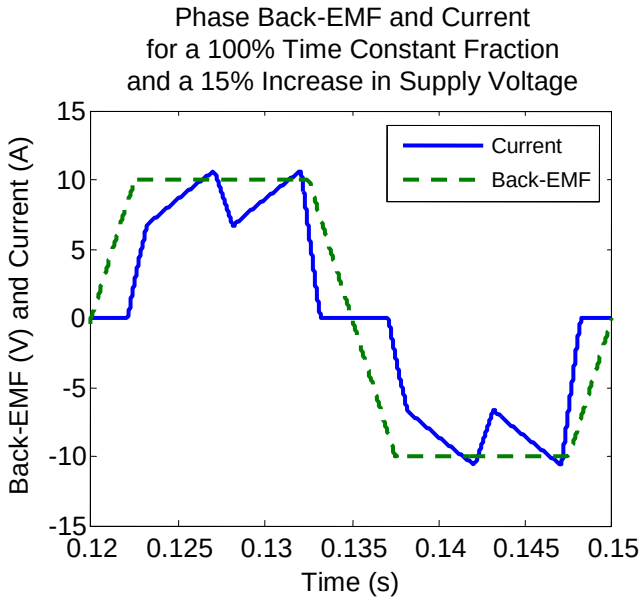


Figure 6.15: Current improvement when using a 15% supply voltage increase.

6.4.3 Characterisation of the Allowable Time Constant

The motor current simulation program was set to calculate the output torque of the motor for various time constant fractions when using the minimum required supply voltage to produce rated torque with no time constant. The results are shown in Figure 6.16.

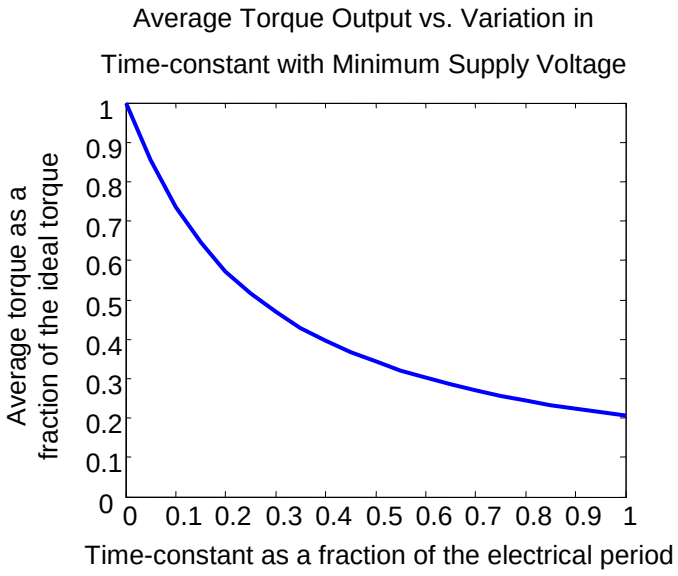


Figure 6.16: Effect of the time constant on torque output when using the minimum supply voltage.

As expected, the rated torque cannot be supplied unless the time constant is extremely small. Practical motor drivers need to use a supply voltage which is slightly above the minimum. Figure 6.17 shows the effect of setting the supply voltage at 1.1 times the minimum value. The result is that rated torque can be supplied with up to a 55% time constant fraction for this motor.

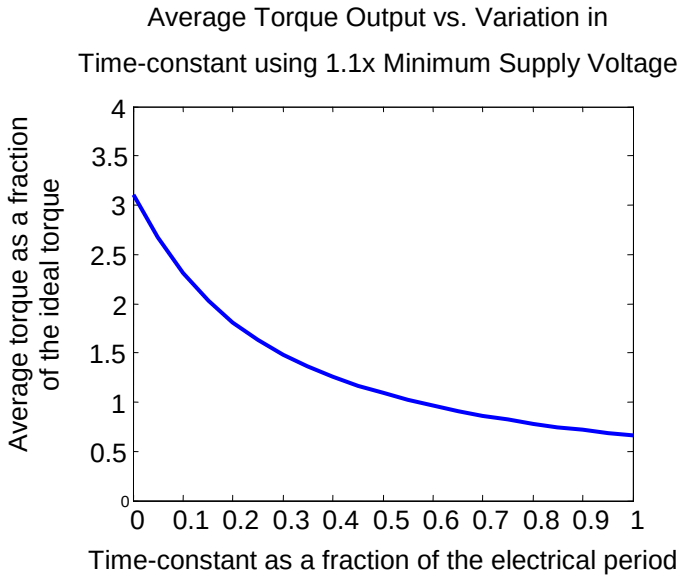


Figure 6.17: Effect of the time constant on torque output when using 1.1x the minimum supply voltage.

The simulations were repeated for a range of supply voltages, and the results are shown in Figure 6.18. In order for this characterisation to apply to any PMBLDC motor with trapezoidal back-EMF, the supply voltage increase could not be based on the minimum required supply voltage. This is because the ratio between the back-EMF and the resistive voltage drop differs between different motors. In order to get around this problem, the steady-state (SS) increase in current (or voltage drop) was used to characterise the increase in the supply voltage. For example, a 2x steady-state current increase factor means that the supply voltage is increased so that the steady-state current would be increased by two times the value it was when using the minimum supply voltage. The term "steady-state" means the level of current that occurs if there was very little or no motor inductance or the phases were not commutated, so the current has time to rise to the SS value.

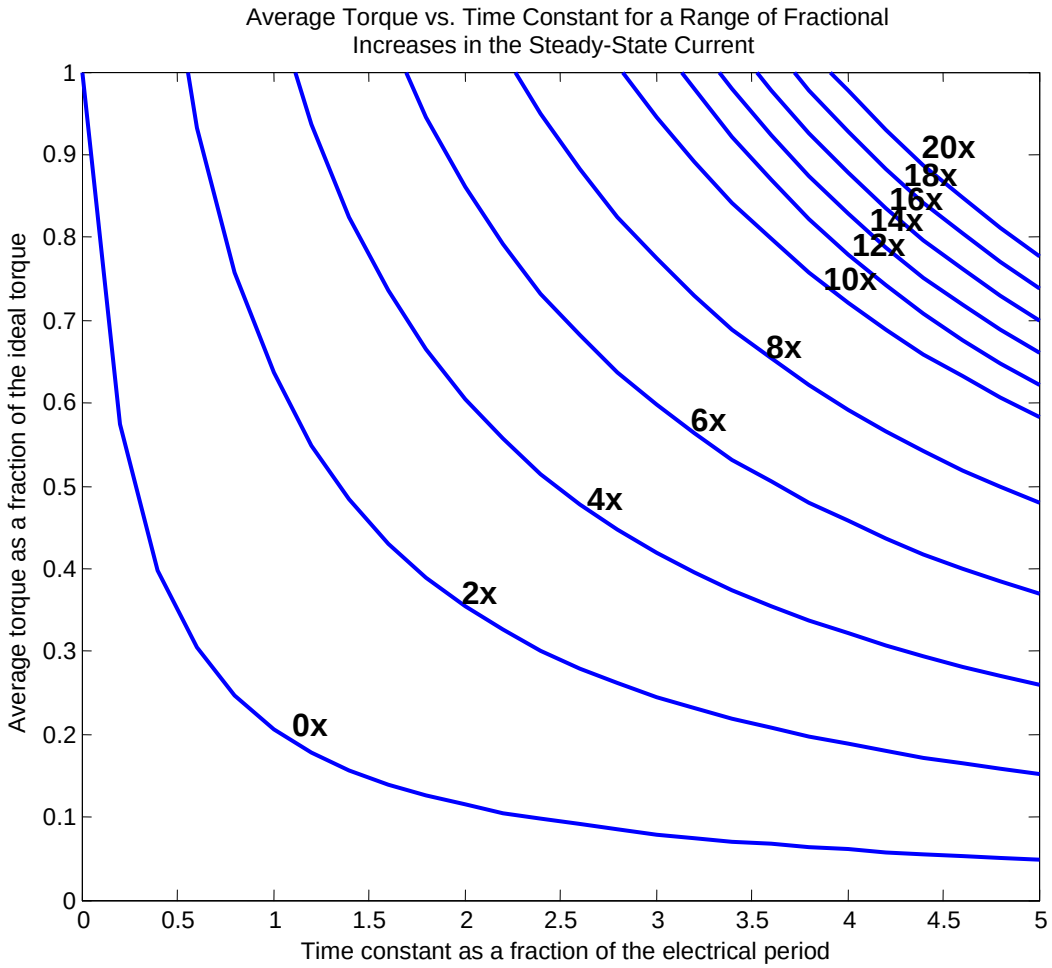


Figure 6.18: Effect of the time constant on the torque output when using a range of supply voltages (characterised by the SS current increases).

The data from Figure 6.18 is used in the PMBLDC motor simulation program in order to determine whether a particular motor design can supply rated torque for a given supply voltage set by the user. The characterisation from this data can be applied to any PMBLDC motor, so long as the back-EMF is approximately trapezoidal. By using this characterisation, full current simulations are not required for each motor simulation. This increases the speed of the motor simulations immensely. If the back-EMF is not trapezoidal, but more sinusoidally shaped, then the results from this characterisation can still be used as a conservative estimation. This is because the trapezoidal back-EMF represents the extreme case, where the back-EMF waveform is already at the peak value when the start of the current commutation occurs.

6.5 Summary

A PMBLDC parametric simulation program was written using MATLAB. The simulation program can be used to optimise a motor for the Electric Jetpack by changing various motor parameters in order to find the optimum power density design. The effects of the motor time constant were investigated, and a characterisation of allowable time constants was made that could be applied to any PMBLDC motor in order to determine if rated torque could be obtained. This was integrated into the PMBLDC motor simulation program as it allows much quicker simulations without the need to simulate full current waveforms for every motor.

Chapter 7

Optimisation Results for a PMBLDC Motor

7.1 Introduction

This chapter covers the results of various preliminary motor designs which are optimised for power density. The types of designs covered are the inside and outside-rotor motors located inside and outside the fan ducts. The choice of motor parameters used in the simulations is also explained.

7.2 Choice of Motor Parameters

The PMBLDC simulation program requires that the user defines the parameters of the simulations. The following sections give an overview of the choice of motor parameters used for the simulations.

7.2.1 Air-gap and Stator Radius

The air-gap radius was constrained to a maximum value of 11 cm for the inside duct motor simulations since the duct size is limited. For the outside duct configuration, the air-gap radius was constrained to a maximum of 14 cm. Higher values were not needed as the optimum motor radius was found to be less than 14 cm from trial simulations with greater ranges. The stator outside radius was constrained to a maximum of 12 cm for the inside duct configuration, and 15 cm for the outside duct configuration. For the outside-rotor designs, the outside radius can not be predefined as this depends on calculations of the rotor yoke width within the simulation. For these motors the inside stator radius is instead used.

7.2.2 Number of Magnet Poles

The number of magnet poles has a direct effect on the power density of the motor. Higher pole counts allow smaller rotor and stator yoke widths, and hence a lighter motor. Increasing the number of poles too much causes the motor time constant

fraction to become too large. The range of magnet poles simulated was from 4-14 poles.

7.2.3 Windings

Concentrated full-pitch windings were used for all integral slot simulations, while the distributed LRK winding was used for the fractional slot variants. Concentrated windings were used since they promote a more trapezoidal back-EMF waveform [11]. The distributed LRK winding was chosen for the fractional slot, 10/14 magnet pole variants. This was because this winding decreases end-turns resistive losses (since each coil is wound over a single tooth). This winding is also a very popular choice for model RC aircraft motors, which often come in the 14 pole, 12 slot PMBLDC type. The fill-factor for the copper windings is approximated to be 40% for all the motor simulations, as per recommendations in [9, 10].

7.2.4 Inside or Outside-Rotor Types

Both the inside and outside-rotor motor variants were tested for the inside duct motor configuration. The inside-rotor motors were shown to achieve a higher power density due to the larger surface area on the stator on which liquid cooling is applied.

7.2.5 Motor Speed

The speed for the inside duct motor simulations was set at 7000 RPM. The speed for the outside duct motor simulations was set to 8235 RPM due to the belt drive ratio. A higher speed can be advantageous for reducing dimensions/increasing power density [4].

7.2.6 RMS Current Density

The RMS current density used for various cooling options is given in Table 7.1. The current density for the motor simulations was chosen to be 20 A/mm^2 as this lies half way through the typical range of current densities for liquid cooled motors. Higher current densities allow less copper to be used. This decreases the lengths of the stator teeth and the outside radius of the motor. These savings allow the weight of the motor to decrease and the power density to increase.

Motor Type	Current Density (A/mm^2)
Totally enclosed	1.5 - 5
Air-over Fan-cooled	5 - 10
Liquid cooled	10 - 30

Table 7.1: Typical current densities for various cooling options. Reproduced from [10].

7.2.7 Magnet Type

Permanent magnets are often compared with their maximum energy product $(BH)_{\max}$. The maximum energy product is calculated by multiplying the magnetic field strength and the flux density together for each point on the demagnetisation operating characteristics, and choosing the greatest value. Table 7.2 gives an overview of the maximum energy product for various magnet types, as well as some other important characteristics.

Magnet Type	Maximum Energy Product (kJ/m^3)	Remanence Flux Density (T)	Maximum Temperature (°C)
Alnico	20 - 100	0.6 - 1.35	500 - 550
Ferrite	24 - 36	0.35 - 0.43	250
Sintered SmCo	140 - 220	0.7 - 1.1	250 - 350
Sintered NdFeB	180 - 320	1.0 - 1.4	80 - 200

Table 7.2: Comparison between different types of permanent magnet material. Data from [10].

The sintered neodymium-iron-boron (NdFeB) magnet offers the largest maximum energy product and the largest remanence flux density. The sintered sarium-cobalt magnet has a lower energy product and remanence flux density, but it can withstand higher temperature operation. The alnico magnet offers good remanence flux density, but it has a very low intrinsic coercivity. This means that the maximum energy product is low, and the magnet is very easily demagnetized.

The sintered NdFeB magnet was chosen to be used for the motor simulations. The NdFeB magnet would allow the greatest air-gap flux density within the motor, and would help to reduce the size and weight of the motor for a given output power.

A survey of different grades of NdFeB magnets from three different suppliers was performed and is shown in Table 7.3.

Magnet Grade	Remanence Flux Density (kG)	Intrinsic Coercivity (kOe)	Maximum Operating Temperature (°C)
Kg Magnetics:			
N38EH	12.2-12.6	>10.8	200
N40UH	12.6-12.9	>11.4	180
N45SH	13.3-13.7	>11.4	150
N48H	13.6-14.2	>11.4	120
Dexter Magnetics:			
N3530	12.0	11.7	200
N3825	12.5	12.2	180
N4221	13.0	12.5	150
N4517	13.4	12.7	120
MCE:			
N3578	12.1	12.0	200
N3575	12.0	11.6	180
N3067	11.0	10.6	150
N4267	13.0	12.6	120

Table 7.3: Comparison between different grades of NdFeB magnets. Data from supplier websites [47-49].

The N3575 grade was chosen for the motor simulations. This was chosen on the basis that it could withstand high temperature operation, and the demagnetization characteristics (as shown in Figure 6.10) showed that the knee region on the 180°C curve was placed at a further negative field strength compared to some of the other grades, which may have looked better based on only Table 7.3 alone.

7.2.8 Core Type and Flux Density

As discussed previously in section 6.2.10, the three lamination materials that were programmed into the simulation program are:

- Silicon-iron standard: M19 0.014 inch
- Silicon-iron thin gauge: Arnon5 Special 0.005 inch
- Cobalt-iron thin gauge: Hyperco 50A 0.006 inch

The silicon-iron thin-gauge core is used for electrical frequencies greater than 400 Hz in order to decrease the core loss of the motor [44]. The cobalt-iron laminations are used in some of the higher frequency simulations in order to determine what scale of power density improvements can be made. The cobalt-iron laminations are expensive compared to the other two lamination materials [45], so they are only recommended if absolutely necessary to meet power density requirements.

The maximum flux density to be used is specified by the user. From inspecting the magnetisation curves of the lamination materials listed above, an acceptable maximum flux density was chosen. For the silicon-iron laminations this was 1.3 T, and for the cobalt-iron laminations it was 1.8 T. These values chosen are slightly lower than the saturation flux density for each material. Solid carbon steel was used for the rotor yoke, and the flux density was limited to 1.5 T according to recommendations in [10].

7.2.9 Turns per Coil

The number of turns used in each coil making up the phase windings determines the back-EMF magnitude. The number of turns also affects the phase inductance and resistance, however it has no effect on the time constant of the windings, providing that the area of copper used in each slot is kept constant. This means that the number of turns can be customised without significant effects on the performance of the motor. When using a low number of turns with large conductor cross-sectional areas, the use of litz wire may be necessary in order to prevent significant AC resistive losses due to eddy current and proximity effects.

7.2.10 Air-gap Length and Magnet Thickness

The air-gap length was chosen to be 1 mm for all simulations. This was chosen on the basis that a kevlar wrap or stainless steel retaining can may be required to hold the magnets in place. For lower speed motors the magnets can be bonded to the rotor with an adhesive [10]. Mechanical analysis has not been included into the simulation program, and so the safer option of including space for a retaining can or kevlar wrap was chosen. The disadvantage of using a larger air-gap is a slight decrease in air-gap flux density. However this can be countered by increasing the thickness of the

magnets. As it turns out, thick magnets were required on the basis of decreasing the air-gap self and mutual inductance between phases for the PMBLDC motor simulations. The magnet thickness was set to 8 mm for all simulations. Thick magnets were not only used to lower the inductance. They were also used since they allow heavy electric loading of the motor without the risk of demagnetisation. The thickness of the magnets may be decreased and optimised using a FEM simulation package if desired, as this allows very accurate determination of demagnetisation.

7.2.11 Magnet Arc Coverage and Tooth Tip Arc Coverage

The magnet arc coverage refers to the fraction of the rotor surface area covered by permanent magnet material. According to the ideal back-EMF waveform that was shown in Figure 6.11, the flat-top regions only needed to be a minimum of 67% of the waveform period (each flat-top region spanning 120 elec. degrees). If no fringing of the magnetic fields occurred, and the motor was wound with star connected full-pitch concentrated windings on an integral slot design, then the magnet arc coverage only needs to be 67% to obtain the 120 elec. degree flat-top regions (from inspecting ideal waveforms in [10]). The advantage of using lower magnet arc coverage is that the total amount of flux in the motor is decreased, and the widths of the stator yoke and rotor yoke can be decreased. In practice, a magnet arc of greater than 67% will be required to account for the fringing of magnetic flux.

The PMBLDC simulation program can be used to show the effects of magnet arc coverage. Figure 7.1 shows an example of the geometry and torque output calculated with the magnet arc set to 100%. Figure 7.2 shows the results of decreasing the magnet arc to 67%. It can be seen that decreasing the magnet arc has allowed a greater torque output, as the area per slot available for windings has increased. The notches in the torque waveforms are due to the slotting. The effects of fringing magnetic flux are not calculated by the PMBLDC simulation program (apart from the approximations of slotting effects), so a magnet arc greater than 67% is required to ensure trapezoidal back-EMF for the integral slot designs. The magnet arc chosen was 80%. This figure is only a conservative approximation. FEM analysis could be used to optimise the magnet arc coverage for a particular design if desired. The FEM

simulations in section 8.2 show that this magnet arc produces a trapezoidal back-EMF that is similar to the ideal back-EMF for a PMBLDC motor.

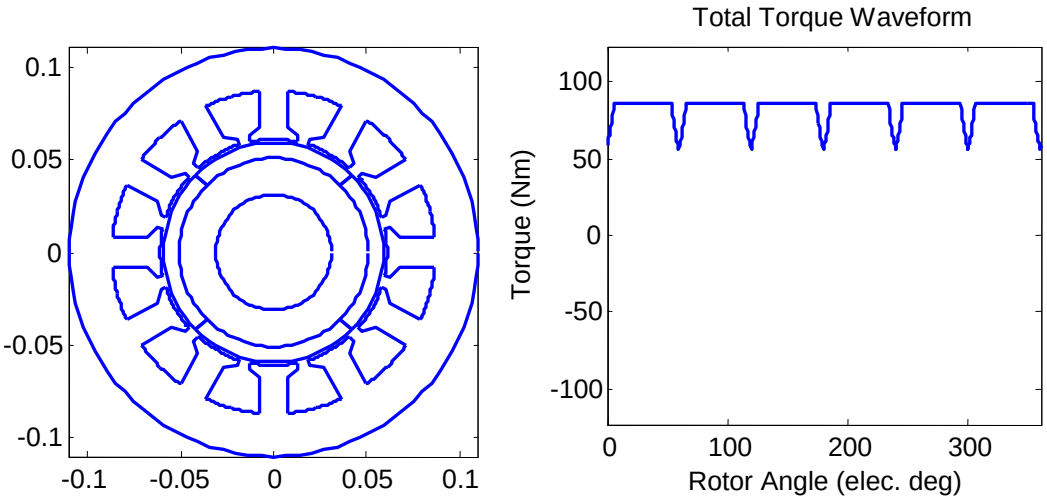


Figure 7.1: Simulation showing motor geometry and torque output for 100% magnet arc coverage.

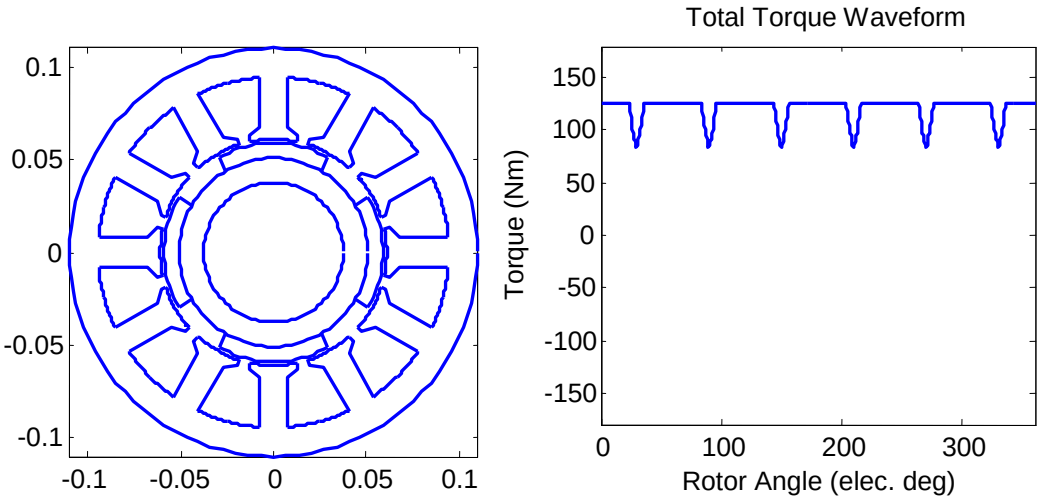


Figure 7.2: Simulation showing motor geometry and torque output for 67% magnet arc coverage.

The tooth-tip arc coverage refers to how wide the tooth-tips are, and determines how much space they leave between the tooth-tips. If the tooth-tip arc is made too large, then there is not enough room to insert the windings into the slots, and the flux will tend to flow around the tooth-tips instead of linking the coils. The slot leakage inductance would also be increased since the flux due to the windings would not have to cross the air-gap, but could flow around the tooth-tips. If the tooth-tip arc coverage is made too small, then the cogging torque of the motor increases [9]. The tooth-tip arc coverage was set to 80% for the motor simulations.

7.2.12 Supply Voltage Factor

For this project, the supply voltage factor refers to the fractional increase in supply voltage over the peak back-EMF. This was limited to 1.5 for these motor simulations. As an example, if the motor peak line-line back-EMF was 100 V, then the supply voltage for the six-step drive was allowed to be a maximum of 150 V. The increase in voltage is used to overcome the effects of the motor time constant. The characterisation of the allowable time constant for a particular increase in supply voltage was determined in section 6.4. For a motor where the steady-state resistive voltage drop is 5% of the peak back-EMF, a supply voltage factor of 1.5 would allow a maximum motor time constant of approximately 300% of the electrical period. The exact value depends on the ratio between the back-EMF and the steady state voltage drop in the winding resistance, and was described in section 6.4. All of the motor designs in the simulation results had time constants less than 300% of the electrical period, and many were less than 200%. Because of this, a supply voltage of 1.5 times the peak back-EMF is not necessary for many of the motor designs simulated.

7.3 Simulation Process

This section details the simulation process used, by example, to find an optimised power density design for a particular motor type. In this example the motor type to simulate is the inside-rotor, 10 pole, 12 slot, distributed LRK type running at 7000 RPM. The purpose of the simulation is to determine the optimum motor geometry in order to maximise power density. Table 7.4 shows the motor parameters for this optimisation.

Motor Variables to Adjust	
Air-gap radius	2 - 11 cm
Outside stator radius	2 - 12 cm
Motor Variables to Remain Constant	
Number of magnet poles	10
Number of stator slots	12
Winding style	Distributed LRK wye
Motor type	Inside rotor
Motor speed	7000 rpm
Motor power	75 kW
RMS current density	20 A/mm^2
Magnet type	Neodymium N3575
Core type	SiFe Arnon5 Special 0.005in
Winding temperature	180°C
Magnet temperature	150°C
Max stator flux density	1.3 T
Max rotor flux density	1.5 T
Turns per coil	10
Slot fill-factor	40%
Air-gap length	1 mm
Magnet thickness	8 mm
Magnet arc coverage	80%
Tooth tip arc coverage	80%
Tooth tip depth	2 mm
Flux leakage	10%
Stacking factor	89%
Supply voltage factor	150%

Table 7.4: Simulation parameters for a 10 pole, 12 slot, inside-rotor motor.

The simulation program was set to sweep each of the variables of interest and store the results so that they may be graphed. The results for the power density are shown in Figure 7.3. These results are not filtered to remove designs which are not practical due to heat-flow, demagnetisation, or geometrical problems. After the results are filtered to remove the problem designs, the attainable power density is significantly lowered. Figure 7.4 shows the power density results after filtering. It can be seen that the maximum power density attainable has been halved.

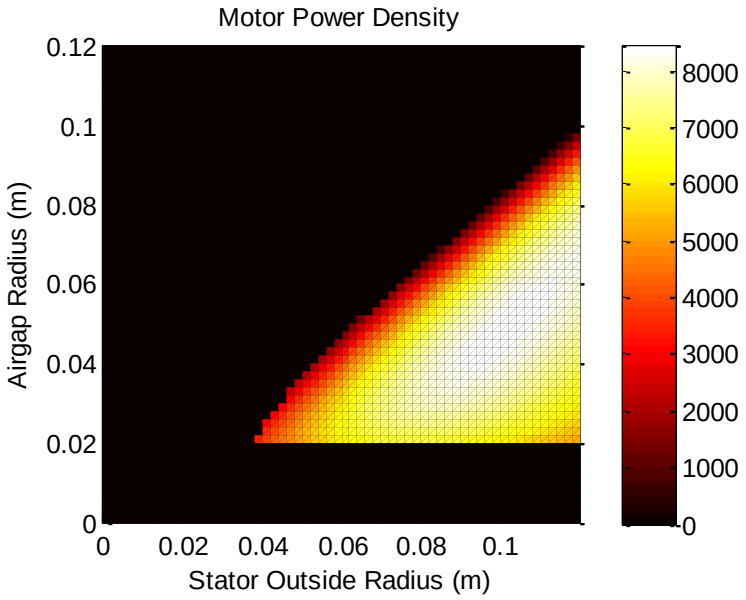


Figure 7.3: Raw simulation results.

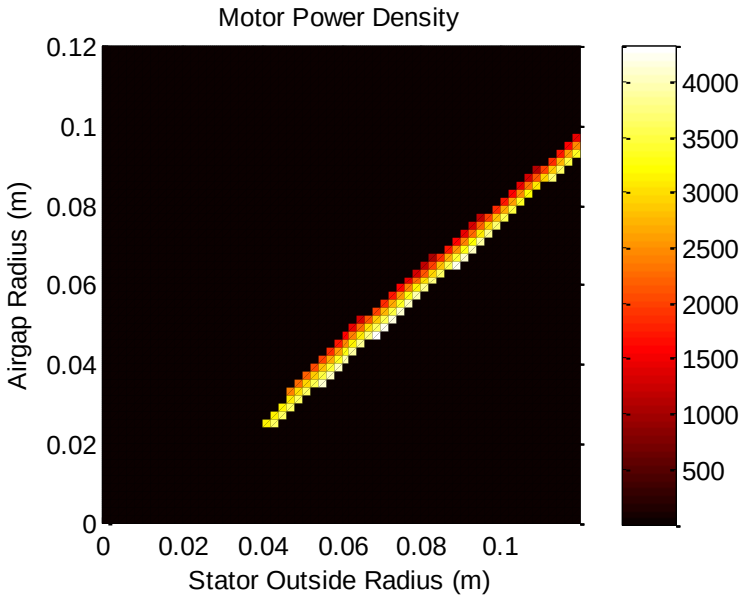


Figure 7.4: Simulation results after filtering problem designs.

The simulation program prints out the motor parameters for the optimum power density design. The optimum motor parameters for this simulation were:

```
Power density: 4333 W/kg
Power: 75000 W
Mass: 17.31 kg
Airgap radius: 46.0 mm
Stator outside radius: 68.0 mm
Axial Length: 248.6 mm
Efficiency: 96.2 %
Rotor yoke width: 6.3 mm
Stator yoke width: 7.1 mm
Tooth width: 14.3 mm
Peak phase back-EMF: 491.4 V
Phase current: 68.2 Arms
Peak phase current: 83.5 A
Average torque: 102.3 Nm
Resistive losses: 2668.2 W
Core losses: 282.0 W
Line-line inductance: 1.07 mH
Line-line resistance: 382.6 mOhm
Time-constant fraction: 1.6
TRV: 61.9 kNm/m^3
```

The last parameter in this list is called the "torque per unit rotor volume" (TRV), and relates closely to the electrical and magnetic loading of the motor [10]. The geometry of the optimum design is graphed, and is shown in Figure 7.5. The geometry is also exported into an AutoCAD file format so that it may be imported into FEM software if desired.

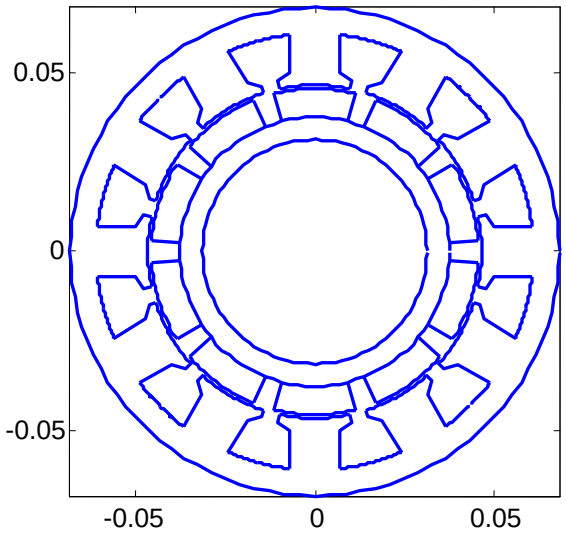


Figure 7.5: Cross-section geometry for an optimised design.

7.4 Summary of Results

The simulation process was repeated for a number of different design options. Parameters that were adjusted included the number of magnet poles, number of slots, winding type, core type, inside and outside-rotor configurations, and inside and outside duct configurations. Table 7.5 shows the optimisation results for the inside-rotor motors located inside the ducts. The best power density achieved was 5.41 kW/kg.

Number of Magnet Poles	4	6	8	10	10	10	14	14
Number of Slots	12	18	24	30	12	12	12	12
Winding Type	Full-pitch	Full-pitch	Full-pitch	Full-pitch	dLRK	dLRK	dLRK	dLRK
Core Type	SiFe 0.35mm	SiFe 0.35mm	SiFe 0.005in	SiFe 0.005in	SiFe 0.005in	CoFe 0.006in	SiFe 0.005in	CoFe 0.006in
Air-gap Radius (mm)	28.0	36.0	36.0	48.0	46.0	50.0	46.0	58.0
Stator Outside Radius (mm)	52.0	58.0	58.0	70.0	68.0	68.0	64.0	74.0
Axial Stack Length (mm)	524.3	383.5	342.4	230.3	248.6	233.5	258.9	199.6
Time constant Fraction	1.1	1.1	1.4	1.5	1.6	1.5	2.0	1.9
Efficiency (%)	94.0	95.2	95.6	96.4	96.2	96.4	96.3	96.7
TRV (kNm/m^3)	79.2	65.5	73.4	61.4	61.9	55.8	59.4	48.5
Power Density (W/kg)	**2777**	**3275**	**4016**	**4412**	**4333**	**4821**	**5197**	**5409**

Table 7.5: Motor simulation results for the inside-rotor, inside of duct configuration.

Table 7.6 shows the results for outside-rotor motors located inside the ducts. The best power density achieved was 3.68 kW/kg.

Number of Magnet Poles	4	6	8	10	10	10	14	14
Number of Slots	12	18	24	30	12	12	12	12
Winding Type	Full-pitch	Full-pitch	Full-pitch	Full-pitch	dLRK	dLRK	dLRK	dLRK
Core Type	SiFe 0.35mm	SiFe 0.35mm	SiFe 0.005in	SiFe 0.005in	SiFe 0.005in	CoFe 0.006in	SiFe 0.005in	CoFe 0.006in
Air-gap Radius (mm)	66.0	62.0	54.0	108.0	86.0	78.0	68.0	68.0
Stator Inside Radius (mm)	30.0	34.0	54.0	78.0	58.0	58.0	48.0	52.0
Axial Stack Length (mm)	482.0	444.5	207.5	141.7	194.0	189.9	235.9	227.4
Time constant Fraction	0.6	0.6	0.9	0.9	1.3	1.2	1.5	1.5
Efficiency (%)	96.8	96.6	97.5	97.5	97.5	97.5	97.4	97.3
TRV (kNm/m^3)	15.5*	19.1*	53.8*	19.7*	22.7*	28.2*	29.9*	31.0*
Power Density (W/kg)	**737**	**1188**	**1764**	**2089**	**2196**	**2896**	**3287**	**3678**

Table 7.6: Motor simulation results for the outside-rotor, inside of duct configuration. *TRV is not usually applied to outside-rotor motors. In this case the air-gap radius has been used, and the TRV values do not actually refer to the rotor volume.

Table 7.7 shows the results for inside-rotor motors located outside the ducts. The best power density achieved was 6.56 kW/kg. Outside-rotor motors located outside the ducts were not simulated. This was because they were shown to have lower power densities during the inside duct simulations.

Number of Magnet Poles	4	6	8	10	10	10	14	14
Number of Slots	12	18	24	30	12	12	12	12
Winding Type	Full-pitch	Full-pitch	Full-pitch	Full-pitch	dLRK	dLRK	dLRK	dLRK
Core Type	SiFe 0.35mm	SiFe 0.005in	SiFe 0.005in	SiFe 0.005in	SiFe 0.005in	CoFe 0.006in	SiFe 0.005in	CoFe 0.006in
Air-gap Radius (mm)	32.0	40.0	44.0	44.0	58.0	48.0	64.0	56.0
Stator Outside Radius (mm)	58.0	66.0	68.0	66.0	82.0	66.0	84.0	72.0
Axial Stack Length (mm)	685.7	464.6	409.5	420.8	287.9	413.0	263.1	348.6
Time constant Fraction	1.5	1.7	1.9	2.0	2.3	1.8	2.9	2.4
Efficiency (%)	95.6	96.5	96.8	96.9	97.3	96.9	97.5	97.2
TRV (kNm/m^3)	78.8	74.5	69.8	67.9	57.2	58.2	51.4	50.6
Power Density (W/kg)	**3401**	**4330**	**4965**	**5413**	**5322**	**5801**	**6339**	**6559**

Table 7.7: Motor simulation results for the inside-rotor, outside of duct configuration.

7.5 Discussion

The power density results do not include the weights of the bearings, cooling jacket, and any other structural and miscellaneous parts, so the actual power density obtained is somewhat lower. The highest power density for the inside duct motors was 5.41 kW/kg, while the highest power density for the outside duct motors was 6.56 kW/kg. These figures correspond to 27.7 kg and 22.9 kg for the total motor weight. Considering that the combined weight of the existing combustion engine and fuel tank is 65 kg, this leaves between 37.3 kg and 42.1 kg of weight for the motor

driver/controller, cable weight, and the parts of the motor weights that were not calculated in the simulations. In section 3.6 it was shown that an approximate weight for the motor driver is 20.8 kg, and in section 5.3.5 it was shown that a 5 meter cable would weigh only 5 kg. After taking these weights into consideration there is still between 11.5 kg and 16.3 kg of weight available for the aluminium cooling jacket, bearings, supports, and any miscellaneous parts. From this basis the tethered Electric Jetpack appears to be feasible, and the available pilot/load weight should not be compromised.

For the inside-rotor motor set inside the ducts the best power density was 5.41 kW/kg. As stated in section 6.2.10, cobalt-iron laminations are used in very demanding applications where weight is a critical factor. When thin gauge silicon-iron laminations were used, the best power density obtainable was approximately 5.20 kW/kg. The extra power density that the cobalt-iron laminations provide may not be worth the extra cost associated with them.

For the outside-rotor motor set inside the ducts the maximum power density was quite lower at 3.68 kW/kg. The main reason the outside-rotor motors did not perform as well as the inside-rotor motors is that the stator surface area available for cooling is less.

For the outside duct motors, the best power density was 6.56 kW/kg. Using thin gauge silicon-iron laminations limited the power density to 6.34 kW/kg. These simulations show a power density improvement over the inside duct motors. This is due to the increased speed of the motors when connected through the belt drive. The improvement over the inside-duct motors is approximately 1.15 kW/kg. For a power output of 150 kW this is a weight saving of only 4.9 kg.

The results also show how increasing the number of magnet poles increases the power density. It was decided to stop at 14 magnet poles based on the simulation results. The time constant fraction gets larger every time the number of magnet poles is increased. This is due to the electrical frequency of the motor increasing, so less time is available to commutate the phase currents. Beyond 14 magnet poles, the time

constant fraction was becoming too large, and it would be less practical to use these motors as the motor driver would require a supply voltage significantly larger than the motor back-EMF in order to obtain rated current/torque. Other disadvantages of increased supply frequency and voltage will be an increase in losses in the motor driver. This requires further study to determine exactly where the number of magnet poles should be set in order to maximise the power density of the whole motor and driver system together. It should also be mentioned that the higher speed motors from the outside duct simulations had larger time constant fractions when compared to the lower speed inside duct motors.

The torque per unit rotor volume (TRV) values for the simulated motors can be used to provide a level of confidence in these power density results. From [10], the expected range of TRV values for aerospace machines is from 30-75 kNm/m^3. It is interesting to see that nearly all the TRV values for the simulation results fall within this range. This should be expected as approximations for liquid cooling heat-flux restrictions were applied to the motor simulations, which has limited the electric loading of the simulated motors.

It is worth discussing why the power density increases for the higher magnet pole, inside-rotor designs, however the TRV value decreases. This effect is due to the change in cross-sectional area that is filled with motor materials. Figure 7.6 shows the motor cross-sections for the 4 pole and the 14 pole motors from the inside duct, inside-rotor simulation results. It can be seen that the 4 pole motor is mostly filled with the iron yoke, magnets, and copper windings, while the 14 pole motor has a large bore in the rotor that is filled with air. While the rotor volume of the 14 pole motor is higher, it has a much larger proportion of empty space compared to the 4 pole motor. The result is that the gravimetric power density has increased, even while the volume of the rotor has increased. The stator yoke in the 14 pole motor is also thinner than the stator yoke in the 4 pole motor. The axial length of the 4 pole motor is twice as long as the axial length of the 14 pole motor.

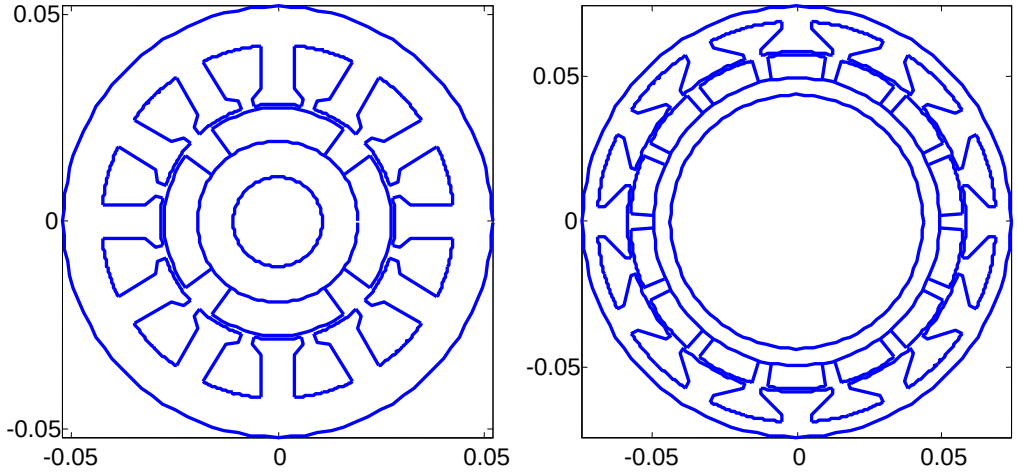


Figure 7.6: Cross-section comparison between the 4 pole and the 14 pole motor.

The main benefit of the fractional slot motors is a decrease in cogging torque [10]. Cogging torque occurs due to the magnet poles aligning with the stator teeth. When they are aligned the reluctance of the magnetic path is at a minimum. Because of this the rotor is attracted to the positions of minimum reluctance. The fractional slot motors have less alignment as the number of magnet poles does not divide integrally into the number of stator teeth.

One benefit of the 12 slot, fractional slot designs is the ability to wind them as fault-tolerant motors. According to [50], the 12 slot, 10/14 magnet pole designs can be wound as 6 phase fault-tolerant motors. In fault-tolerant motors, the phases are separated magnetically, physically, and electrically. Figure 7.7 shows an example of the phase coil placement in a 10 pole, 12 slot, fault-tolerant motor.

Figure 7.7: Phase coil placements in a six phase, 10 pole, 12 slot, fault-tolerant motor. Reproduced from [50].

Each phase has a single coil wound over a single tooth, and an unwound tooth is left between each coil in order to separate the magnetic flux paths. The result is that there is no mutual inductance between the phases [51]. Each phase is controlled by a separate H-bridge so that they are electrically independent of each other. If one phase faults (by open or short circuit), then the other phases can be over-driven to compensate for the lost torque. The fault-tolerant winding is worth considering for the Electric Jetpack as an added safety feature. The disadvantage is that four switching devices are needed for each phase [51], while in the traditional six-step drive only six switches are needed for all three phases.

7.6 Summary

The motor parameters used for the simulations were described. A number of simulations were made to produce some preliminary optimised motor designs for a range of possible parameters. These included the placement of the motor, inside and outside-rotor types, the core material, the number of magnet poles, and fractional slot and integral slot designs. The best power density result for the inside duct motors was 5.41 kW/kg, while the best result for the outside-duct motors was 6.56 kW/kg. These required cobalt-iron laminations. The use of thin gauge silicon-iron laminations decreased the maximum power density slightly. When using the best power density results, the tethered Electric Jetpack should be feasible, and it was shown that the pilot weight would not be compromised. The simulation results were confirmed to fall within the expected TRV range for aerospace machines. It was shown that the higher pole number motors could obtain lower TRV values and higher gravimetric power densities due to the larger proportion of free space in these motors. The fractional slot motors have lower cogging torque, and the ability to be wound as fault-tolerant designs.

Chapter 8

FEM Analysis

8.1 Introduction

The PMBLDC motor parametric simulation program outputs the motor geometry into an AutoCAD file. This makes the task of setting up a FEM simulation of the motor much easier. In this chapter one of the designs from the PMBLDC motor simulation program is compared with the results of a FEM simulation.

8.2 FEM Results

The motor chosen to test was the 6 pole, 18 slot, inner-rotor motor that was optimised for power density in the previous section. A printout of the parameters found by the PMBLDC motor simulation program is given below:

```
Power density: 3275 W/kg
Power: 75000 W
Mass: 22.90 kg
Air-gap radius: 36.0 mm
Stator outside radius: 58.0 mm
Axial Length: 383.5 mm
Efficiency: 95.2 %
Rotor yoke width: 7.9 mm
Stator yoke width: 8.4 mm
Tooth width: 7.0 mm
Peak phase back-EMF: 428.6 V
Phase current: 72.9 Arms
Peak phase current: 89.3 A
Average torque: 102.3 Nm
Resistive losses: 3319.7 W
Core losses: 460.0 W
Line-line inductance: 1.33 mH
Line-line resistance: 416.5 mOhm
Time-constant fraction: 1.1
TRV: 65.5 kNm/m^3
```

The AutoCAD geometry was imported into “Infolytica MagNet 6.23” FEM analysis software. The material for the stator yoke was set to silicon iron M19 0.35 mm laminations, and the material for the rotor yoke was set to carbon steel. The material definition for the N3575 neodymium magnets had to be added. This was done by customizing a pre-existing material definition for a generic neodymium magnet. The intrinsic coercivity was set to -732112 A/m by inspection of the demagnetization characteristics at the target operating temperature. The relative recoil permeability was set to 1.05. The maximum mesh size was set to 1 mm. Figure 8.1 shows the results of the mesh generation.

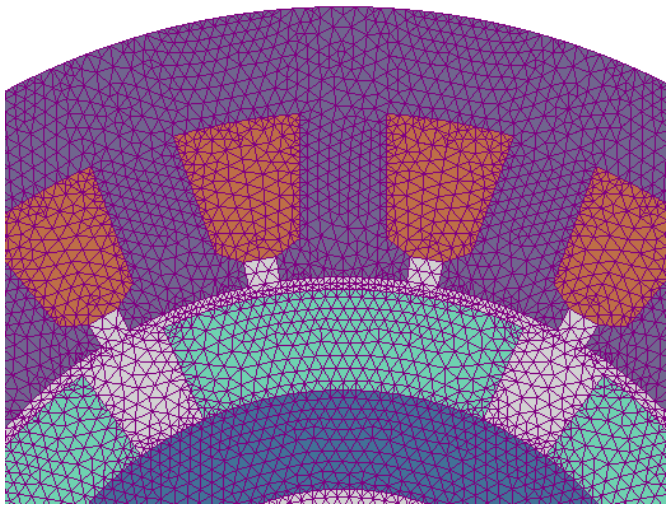


Figure 8.1: FEM mesh generation.

The first parameter to test was the winding inductance. To do this, two phase windings were defined as a single winding in series (to simulate a star connected PMBLDC motor), and the peak current of 89.3 A was used. The flux linkage of the winding was determined and used to calculate the line-line inductance. Figure 8.2 shows the results of the FEM simulation. The magnet flux is disabled when calculating the inductance. It can be seen that there is very little armature reaction flux in the rotor yoke. This helps to justify the use of a solid carbon steel yoke for the rotor.

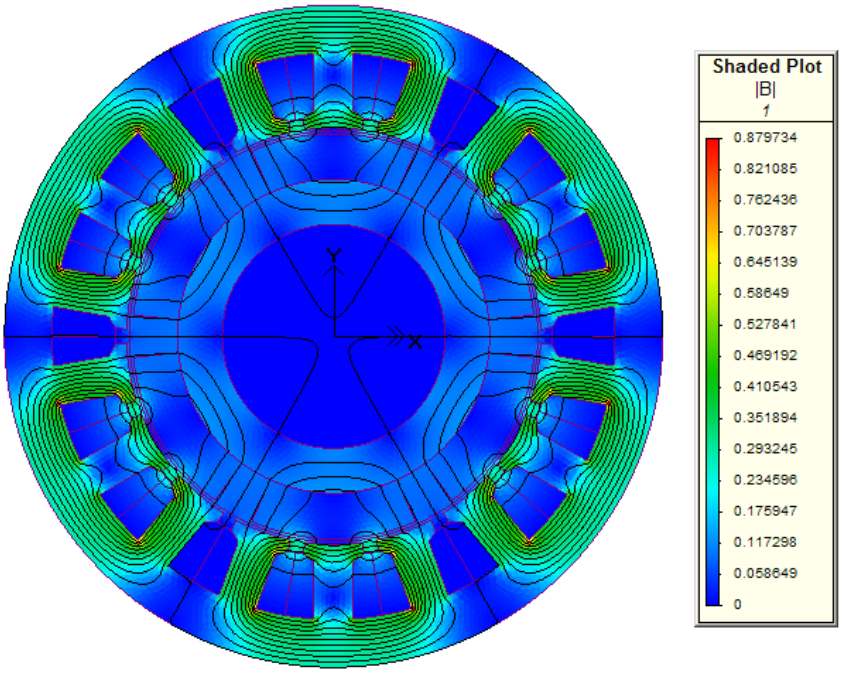


Figure 8.2: FEM flux density plot from windings acting alone.

The calculation for the line-line inductance is

$$L_{ll} = \frac{\lambda}{i} = \frac{0.1186}{89.3} = 1.33\ mH \qquad (8.1)$$

where λ is the flux linkage and i is the current. This result is the same as the parametric result. This does not mean the parametric model is 100% accurate (and it shouldn't be expected to be). The FEM simulation does not include the end-turns inductance, so in actual fact the parametric model has produced an inductance result which is too low. If the end-turns inductance is subtracted from the parametric result, the parametric line-line inductance comes to be 1.29 mH. This is still close to the FEM value.

Figure 8.3 shows the flux density FEM simulation when the magnets are activated and the winding current is deactivated. The flux density in the stator yoke measured 1.32 T, while the flux density in the rotor yoke measured 1.44 T. The parametric goals for the flux density were 1.30 T for the stator yoke and 1.50 T for the rotor yoke. These results are reasonably accurate, given all the approximations that are made in the parametric simulations.

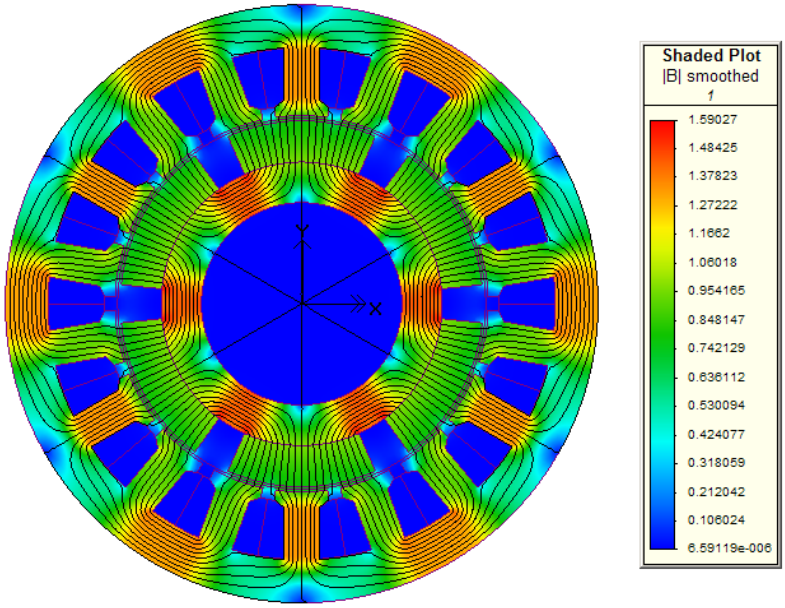


Figure 8.3: FEM flux density plot from magnets acting alone.

The phase back-EMF waveform from the parametric simulation is shown in Figure 8.4. It can be seen that the back-EMF transitions slope steeply. A FEM simulation should produce more shallow sloped transitions as it accounts for the flux fringing and yoke reluctance. The FEM software was used to compute the back-EMF by measuring the flux linkage of the magnetic flux into a phase winding for a range of rotor angles. The back-EMF was determined by differentiation of the flux linkage waveform with respect to time. Figure 8.5 shows the computed back-EMF waveform. The two waveforms roughly agree with each other in terms of the flat-top regions of the back-EMF. Since the flat-top regions are similar, the parametric result will produce a good approximation of the average motor torque (as current is driven into the phase during the flat-top potions of the back-EMF). The predicted slotting effects were too large in the parametric simulation.

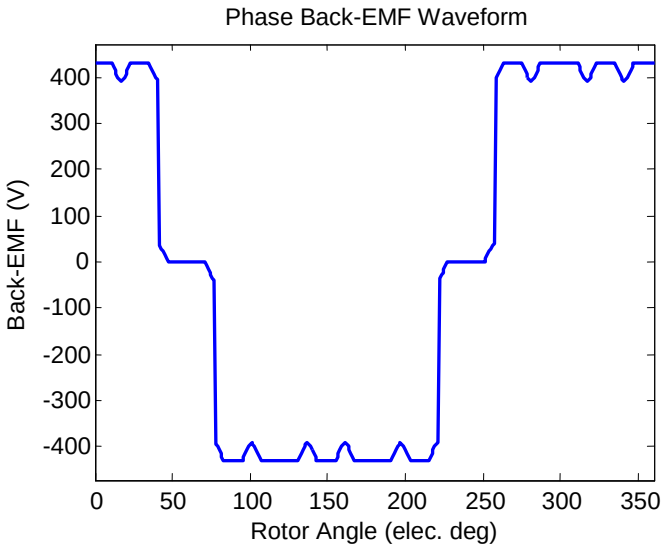


Figure 8.4: Parametric phase back-EMF simulation for a 6 pole, 12 slot motor.

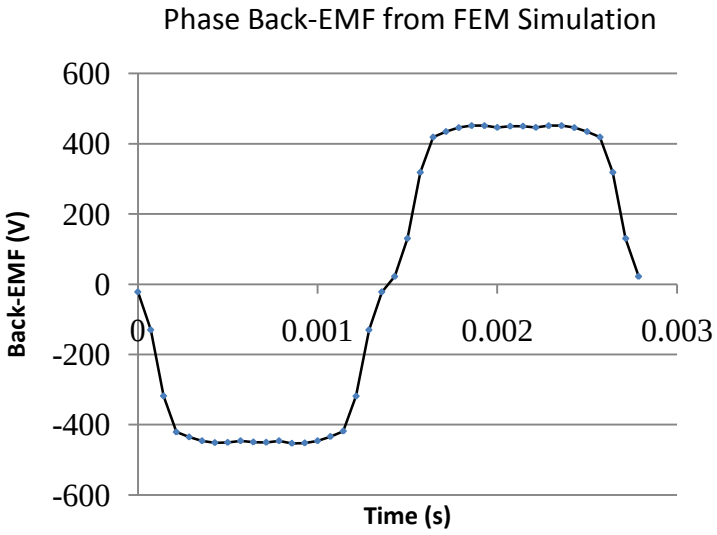


Figure 8.5: Phase back-EMF from FEM simulation.

8.3 Discussion and Summary

The FEM simulations confirmed the results from the parametric simulations. There is room for improvement with some areas. The slotting effects in the back-EMF were over-estimated and the transitions are too steep.

Chapter 9

Future Work and Conclusions

9.1 Future Work

If the electric jetpack is further developed, it will be for the purpose of creating a tethered Electric Jetpack as current energy storage technologies are unsuitable. Before a prototype motor can be built, the mechanical side of the motor design has to be determined. Whether the rotor magnets require a kevlar wrap or stainless steel retaining ring has not yet been determined. The type of bearings to use and shaft dimensions is also not yet determined. The motor driver also needs to be designed or purchased.

The PMBLDC motor simulations showed that the time constant for these motors puts some restrictions on the maximum electrical frequency for the motor. This means that an investigation into the PMBLAC motor would be a worthwhile pursuit. While the PMBLAC motor has a slightly lower torque output in theory, it can cope with larger time constants as the currents do not need to transition sharply as they do in the PMBLDC motor. The PMBLAC motor also responds better to flux weakening operation (including phase advance techniques) [52, 53]. The PMBLDC simulation program could be modified to include of the option of simulating the PMBLAC motor.

There is opportunity to make some slight improvements on the PMBLDC simulation program based on the FEM results. An investigation into the effects of the motor electrical frequency and the supply voltage on the power density of the motor driver would be worthwhile. This would allow the optimum number of magnet poles to be accurately determined as the motor and driver could be analysed together. The use of fault-tolerant windings and a fault-tolerant motor driver should be considered, and the use of a step-down voltage converter could be investigated, as there may be opportunity to make further weight savings in the tethered power cable.

9.2 Conclusions

The feasibility of an electrically powered Jetpack has been determined by research into energy storage technologies, simulations of flight times, surveys of currently available electric motors, and the development of a simulation program which was used to simulate some preliminary optimised motor designs. The overall conclusion of this project is that the Electric Jetpack should be feasible when using a tethered power cable to power the Electric Jetpack, but it is not currently feasible when using on-board energy storage (although very short flight times may still be obtained).

A number of energy storage technologies were investigated. These were batteries, fuel cells, and ultra-capacitors. The lithium iron nano-phosphate battery could power the Electric Jetpack for only a few minutes, with no reserve power capability for manoeuvrability, different air densities, or heavier pilot weights. All the other energy storage technologies were unsuitable. Advancements being made in energy and power density for the lithium-ion battery showed that this may be a viable energy storage technology in the future. The tethered Electric Jetpack was shown to be a feasible as an alternative to using on-board energy storage. The best cable to use is a high strand count flexible copper cable using a high DC bus voltage. The cable was shown to weight as little a 5 kg for a 5 meter cable when using a 1000 V bus. The carbon nano-tube based cable may be a future technology which could further lighten the power cable.

The torque-speed and power density requirements for the electric motor were determined. A large number of motors were surveyed, however none of the motors met both the power density and speed requirements for the Electric Jetpack.

The PMBLDC motor was chosen as the most suitable motor type for the Electric Jetpack due to it having the highest theoretical power density (although it was determined that the PMBLAC motor should be investigated also since it has less issues with the motor time constant). A simulation program was developed that was used to simulate and optimise preliminary high power density motors for the Electric Jetpack. The effects of the motor time constant were investigated, and a characterisation of allowable time constants was made. This was integrated into the

PMBLDC motor simulation program as it allowed much quicker simulations without the need to simulate the current waveforms. The optimised motor designs were shown to achieve a power density of 5.41 kW/kg for the inside duct motor and 6.56 kW/kg for the outside duct motor. These power density results did not include the weight of the bearings, cooling jacket, structural supports, or any other miscellaneous parts. Therefore, the actual power density obtained would be slightly lower than these simulations show. By taking into account the expected weights for the motor driver and power cable, it was shown that the tethered Electric Jetpack should be feasible without compromising the allowable pilot/load weight.

References

[1] M. A. Company. (2010). *Martin Jetpack Photo Gallery*. [Online]. Available: http://www.martinjetpack.com/photo-gallery.aspx

[2] H. F. a. E. Society. (2008, Nov. 19). *Hybrid Cars Too Quiet For Pedestrian Safety? Add Engine Noise, Say Researchers*. [Online]. Available: http://www.sciencedaily.com/releases/2008/11/081117091633.htm

[3] M. A. Company. (2010). *Technical Information*. [Online]. Available: http://www.martinjetpack.com/technical-information.aspx

[4] J. F. Gieras, *Advancements in electric machines*. New York: Springer, 2009. p. 81.

[5] M. Zeraoulia, *et al.*, "Electric Motor Drive Selection Issues for HEV Propulsion Systems: A Comparative Study," *Vehicular Technology, IEEE Transactions on*, vol. 55, pp. 1756-1764, 2006.

[6] A. T. d. Almeida, *et al.* (2008). EUP Lot 11 Motors Final Report. p. 35. [Online]. Available: http://www.ecomotors.org/files/Lot11_Motors_FinalReport.pdf

[7] G. Passier, *et al.*, "Status Overview of Hybrid and Electric Vechile technology (2007)," International Energy Agency, 2007.

[8] R. Krishnan, *Permanent magnet synchronous and brushless DC motor drives*. Boca Raton: CRC Press/Taylor & Francis, 2010.

[9] D. C. Hanselman, *Brushless permanent magnet motor design*, 2nd ed. Cranston, R.I.: The Writers' Collective, 2003. pp. 67-115, 143-148, 158-176, 183-200, 217-223.

[10] J. R. Hendershot and T. J. E. Miller, *Design of brushless permanent-magnet motors*. Hillsboro, Ohio: Magna Physics, 1994. pp. 3.1-5.59, 8.1-10.30, 15.1-16.13.

[11] S. W. Colton, "Design and Prototyping Methods for Brushless Motors and Motor Control," Master of Science, Mechanical Engineering, Massachusetts Institute of Technology, 2010.

[12] T. J. E. Miller and R. Rabinovici, "Back-EMF waveforms and core losses in brushless DC motors," *Electric Power Applications, IEE Proceedings -*, vol. 141, pp. 144-154, 1994.

[13] D. C. Hanselman, "Effect of skew, pole count and slot count on brushless motor radial force, cogging torque and back EMF," *Electric Power Applications, IEE Proceedings -*, vol. 144, pp. 111-113, 325-330, 1997.

[14] D. A. Staton and A. Cavagnino, "Convection Heat Transfer and Flow Calculations Suitable for Electric Machines Thermal Models," *Ieee Transactions on Industrial Electronics*, vol. 55, pp. 3509-3516, Oct 2008.

[15] UQM Technologies. (n.d.). *Vehicle Propulsion Systems*. [Online]. Available: http://www.uqm.com/propulsion_specs.php

[16] CalMotors. (n.d.). *Products*. [Online]. Available: http://www.calmotors.com/products.php?category=products&page=main

[17] Evo Electric Ltd. (n.d.). *Electric Motors: Products*. [Online]. Available: http://www.evo-electric.com/products/electric-motors/

[18] University of Waterloo Microelectronics Heat Transfer Laboratory. (1997). *Fluid Properties Calculator*. [Online]. Available: http://www.mhtl.uwaterloo.ca/old/onlinetools/airprop/airprop.html

[19] T. Muneer, *et al.*, *Heat transfer : a problem solving approach*. London ; New York: Taylor & Francis, 2003. p. 168.

[20] Raser Technologies. (2010). *PDFs*. [Online]. Available: http://www.rasertech.com/pdfs

[21] Y. International. (n.d.). *Power Drive System Specifications*. Available: http://yuneeccouk.site.securepod.com/PowerMotor_Tech_spec.html

[22] D. Linden and T. B. Reddy, *Handbook of batteries*, 3rd ed. New York: McGraw-Hill, 2002. p. 3.2, 22.12.

[23] J. Axsen, *et al.*, "Batteries for Plug-in Hybrid Electric Vehicles (PHEVs): Goals and the State of Technology circa 2008," University of California, UCD-ITS-RR-08-14, May 2008.

[24] A123 Systems. (n.d.). *Products*. [Online]. Available: http://www.a123systems.com/a123/products

[25] J. Hong, *et al.*, "Vanadium Modified LiFePO4 Cathode for Li-Ion Batteries," *Electrochemical and Solid State Letters*, vol. 12, pp. A33-A38, 2009.

[26] Enerdel. (n.d.). *Design*. [Online]. Available: http://www.ener1.com/?q=content/enerdel-design

[27] C. K. Chan, *et al.*, "High-performance lithium battery anodes using silicon nanowires," *Nature Nanotechnology*, vol. 3, pp. 31-35, Jan 2008.

[28] F. R. Kalhammer, *et al.*, "Status and Prospects for Zero Emissions Vehicle Technology, Report of the ARB Independent Expert Panel 2007," California Air Resources Board, Sacramento, California.2007.

[29] C. E. Thomas, "Fuel cell and battery electric vehicles compared," *International Journal of Hydrogen Energy*, vol. 34, pp. 6005-6020, Aug 2009.

[30] U.S. DOE Energy Efficiency and Renewable Energy (2009, June 6). *FCT Fuel Cells: Types of Fuel Cells*. [Online]. Available: http://www1.eere.energy.gov/hydrogenandfuelcells/fuelcells/fc_types.html

[31] A. Burke, "Ultracapacitor technologies and application in hybrid and electric vehicles," *International Journal of Energy Research*, vol. 34, pp. 133-151, Feb 2010

.

[32] E. J. Lerner. (2004, Oct 21). *Less is more with aerogels*. [Online]. Available: http://www.aip.org/tip/INPHFA/vol-10/iss-5/p26.html

[33] APowerCap Technologies. (n.d.). *Ultracapacitors and Solutions / Products*. [Online]. Available: http://www.apowercap.com/?pg=18&lang=eng&rand=12960580

[34] Cooper Bussmann. (n.d.). *PowerStor Supercapacitors B Series Datasheet*. [Online]. Available: http://www.cooperbussmann.com/3/CooperPowerStorDataSheetList.html

[35] D. R. Lide, *CRC Handbook of Chemistry and Physics*, 84th ed. Boca Raton, FL: CRC Press, 2003.

[36] W. A. Thue, *Electrical power cable engineering*, 2nd ed. New York: Marcel Dekker, 2003. pp. 2.2-2.6

[37] Aerco. (n.d.). *HI-FLEX Power Cable*. [Online]. Available: http://www.aerco.co.uk/Cable_Wire_Accessories__Habia+Cable_HIFLEX+Power+Cable.htm

[38] General Cable. (n.d.). *Flexible Cables*. [Online]. Available: http://www.generalcable.co.nz/newzealand/Products/Energy/Flex_Cable.aspx

[39] Storm Products Company. (n.d.). *Cable Interconnect "System" Design and Component Selection Manual*. [Online]. Available: http://www.stormcable.com/uploads/Wire_conductors.pdf

[40] American Superconductor. (2010). *Second Generation (2G) HTS Wire*. [Online]. Available: http://www.amsc.com/products/htswire/2GWireTechnology.html

[41] J. Hartsfield and J. Boyd. (2005, April 5). *NASA Awards $11 Million Quantum Wire Contract to Rice*. [Online]. Available: http://www.nasa.gov/centers/johnson/news/releases/J05-018_prt.htm

[42] L. Retzbach. (2001) Performance Measurements on Electric Motors, DIY Motor Test Rig. *ElektroModell*.

[43] H. Weh, *et al.*, "New permanent magnet excited synchronous machine with high efficiency at low speeds," in *Proc. of the ICEM*, 1988, pp. 35-40.

[44] Steve Constantinides, "Designing with Thin Gauge," in *SMMA Fall Technical Conference*, St. Louis, 2008.

[45] Proto Laminations Inc. (n.d.). *Cobalt Alloys for Laminations*. [Online]. Available: http://www.protolam.com/page9.html

[46] MCE Engineering. (n.d.). *Neodymium Iron Boron (Sintered) N3575*. [Online]. Available: http://www.mceproducts.com/materials/material-details.asp?id=1008&MaterialTypeCode=1003

[47] K&J Magnetics. (n.d.). *Neodymium Magnet Physical Properties*. [Online]. Available: http://www.kjmagnetics.com/specs.asp

[48] Dexter Magnetics. (n.d.). *Neodymium Iron Boron (Nd-Fe-B) Magnet Material Grades*. [Online]. Available: http://www.dextermag.com/Neodymium-Iron-Boron-Magnets

[49] MCE Engineering. (n.d.). *Materials / Neodymium Iron Boron (Sintered)*. [Online]. Available: http://www.mceproducts.com/materials/material-list.asp?MaterialTypeCode=1003

[50] K. Atallah, *et al.*, "Torque ripple minimisation in modular permanent magnet brushless machines," in *Electric Machines and Drives Conference, 2003. IEMDC'03. IEEE International*, 2003, pp. 370-375 vol.1.

[51] B. C. Mecrow, *et al.*, "Fault-tolerant permanent magnet machine drives," *Electric Power Applications, IEE Proceedings* -, vol. 143, pp. 437-442, 1996.

[52] Y. F. Shi, *et al.*, "Torque-speed characteristics of interior-magnet machines in brushless AC and DC modes, with particular reference to their flux-weakening performance," *Ipemc 2006: Ces/Ieee 5th International Power Electronics and Motion Control Conference, Vols 1-3, Conference Proceedings*, pp. 1847-1851, 2006.

[53] Z. Q. Zhu, *et al.*, "Flux-weakening characteristics of trapezoidal back-EMF machines in brushless DC and AC modes," *Ipemc 2006: Ces/Ieee 5th International Power Electronics and Motion Control Conference, Vols 1-3, Conference Proceedings*, pp. 908-912, 2006.

Appendix A:

Motor Survey Results

	Motor Type	Motor Weight	Nom Power @7000 RPM	Peak Power @ 7000 RPM	Best Nom. Power	Speed at Nom. Power
		kg	kW	kW	kW	RPM
UQM HiTor	PMBL	41	n/a	n/a	30	?
UQM PowerPhase 75	PMBL	41	20	75	45	4000
UQM PowerPhase 100	PMBL	91	n/a	n/a	100	3000
UQM PowerPhase 125	PMBL	41	?	125	45	3000
UQM PowerPhase 145	PMBL	50	55	145	85	5000
UQM PowerPhase 150	PMBL	91	n/a	n/a	100	3000
TM4 Motive MO120	PMBL	26	37	120	37	7000 – 10000
Calmotors GP150WC	PMBL	44	43	43	43	7000
Calmotors GP200WC	PMBL	50	57	83	59	7800
Calmotors GP250WC	PMBL	55	79	159	96	11000
Calmotors GP750WC	PMBL	143	n/a	n/a	151	4600
EVO-Electric AF-140	PMBL	40	n/a	n/a	72	3600
EVO-Electric AF-240	PMBL	84	n/a	n/a	145	3600
AC Propulsion AC-150	Induction	50	50	150	50	7000
Enova EDU 90	Induction	65	?	?	30	?
LeTourneau SR 200	SRM	230	30	?	30	1800 – 8000
Siemens 1PV5135-4WS28	Induction	90	67	120	67	4000 - 10000
Siemens 1PV5138-4WS24	Induction	120	85	135	85	5000 - 8000
BRUSA HSM6.17.12	Hybrid Synch.	53	45	75	45	6000 - 8000
MES-DEA 200-330W	Induction	79.5	?	?	40	2850
Raser P-200	Induction	112	95	190	100	5000 - 10000
Guang Dong M&C	PMBL	136	68	100	68	7000
DPMB M&C	PMBL	60	n/a	n/a	50	3500
AIM-55 M&C	Induction	185	n/a	n/a	55	4000
Enova EDM 240	Induction	290	?	?	150	?
Enova EDM 120	Induction	160	?	?	65	?
Yuneec Power Drive 400	PMBL	17.5	n/a	n/a	40	?
Strecker Motoren 990.8	PMBL	7.5	n/a	n/a	30	?
AeroConversions	PMBL	22.7	?	?	37	?
SkySpark	PMBL	40	n/a	n/a	60	2500
PMG 132	PMBL	11.2	n/a	n/a	4.5	?
EM42	PMBL	29.1	n/a	n/a	38.5	?

Table A.1: Motor survey results part 1/3.

	Peak Efficiency	**Cooling**	**Min. Coolant Flow Rate**	**Max Speed**	**Length**	**Diameter to Enclose Motor**	**Time at Peak Power**
	%	Air or Liquid	Litres/min	RPM	mm	mm	seconds
UQM HiTor	93	L	8	6500	241	405	45 – 60
UQM PowerPhase 75	94	L	8	8000	252	280	30 – 90
UQM PowerPhase 100	90	L	8	5000	241	40.5	45 – 60
UQM PowerPhase 125	94	L	8	8000	252	280	30 – 90
UQM PowerPhase 145	94	L	8	8000	279	280	30 – 90
UQM PowerPhase 150	95	L	8	5000	241	405	30 – 90
TM4 Motive MO120	97	L	6.7	10000	347	260	10
Calmotors GP150WC	97	L	5	8000	325	≈220	?
Calmotors GP200WC	97	L	5	8800	376	≈220	?
Calmotors GP250WC	97	L	5	13900	427	≈220	?
Calmotors GP750WC	98	L	5	6300	483	≈300	?
EVO-Electric AF-140	96	L	10	5000	115	380	45
EVO-Electric AF-240	96	L	10	5000	234	380	45
AC Propulsion AC-150	91	A	?	13000	350	245	?
Enova EDU 90	?	L	12	10000	?	?	?
LeTourneau SR 200	?	A	?	8000	681	350	?
Siemens 1PV5135-4WS28	?	L	16	10000	425	346	60
Siemens 1PV5138-4WS24	?	L	?	9000	510	346	60
BRUSA HSM6.17.12	96	L	?	11000	245	270	30
MES-DEA 200-330W	?	L	?	9000	471	235	?
Raser P-200	93	L	?	10000	457	279	?
Guang Dong M&C	?	L	?	8000	392	300	?
DPMB M&C	96.3	?	?	?	?	?	?
AIM-55 M&C	94.2	A	?	6000	?	?	?
Enova EDM 240	?	L	16	7200	?	?	?
Enova EDM 120	?	L	16	7200	?	?	?
Yuneec Power Drive 400	?	A	?	?	?	240	?
Strecker Motoren 990.8	?	A	?	6840	160	120	?
AeroConversions	> 90	L	?	?	?	?	?
SkySpark	?	L	?	?	500	350	180
PMG 132	88.6	A	?	2380	200	220	?
EM42	90	A	?	1800	272	250	?

Table A.2: Motor survey results part 2/3.

	Power Density kW/kg	**Peak Power Density** kW/kg	**Controller Weight** kg	**Number for 75 kW @ 7000 RPM**	**Number for 150 kW**	**Motor Weight as Single Drive** kg	**Motor Weight as Duel Drive** kg
UQM HiTor	0.73	1.22	15.9	n/a	5	205	n/a
UQM PowerPhase 75	1.1	1.83	15.9	4	4	164	328
UQM PowerPhase 100	1.1	1.1	15.9	n/a	2	182	n/a
UQM PowerPhase 125	1.1	3.05	15.9	?	4	164	n/a
UQM PowerPhase 145	1.7	2.9	15.9	2	2	100	200
UQM PowerPhase 150	1.1	1.65	15.9	n/a	2	182	n/a
TM4 Motive MO120	1.42	4.62	42	2	4	104	104
Calmotors GP150WC	0.98	1.66	16	2	4	176	176
Calmotors GP200WC	1.18	2.24	16	2	3	150	200
Calmotors GP250WC	1.75	3.76	16	1	2	110	110
Calmotors GP750WC	1.06	1.41	36	n/a	1	143	n/a
EVO-Electric AF-140	1.8	2.75	16	n/a	2	80	n/a
EVO-Electric AF-240	1.73	2.62	16	n/a	1	84	n/a
AC Propulsion AC-150	1	3	30	2	4	200	200
Enova EDU 90	0.46	1.38	35	?	5	325	n/a
LeTourneau SR 200	0.13	?	?	3	5	1150	1380
Siemens 1PV5135-4WS28	0.74	1.33	65	2	3	270	360
Siemens 1PV5138-4WS24	0.71	1.25	65	1	2	240	240
BRUSA HSM6.17.12	0.85	1.55	9.5	2	4	212	212
MES-DEA 200-330W	0.5	?	7.5	?	4	318	n/a
Raser P-200	0.89	1.79	36	1	2	224	224
Guang Dong M&C	0.5	0.74	30	2	3	408	544
DPMB M&C	0.83	1.67	?	n/a	3	180	n/a
AIM-55 M&C	0.3	0.59	30	n/a	3	555	n/a
Enova EDM 240	0.52	0.83	41	?	1	290	n/a
Enova EDM 120	0.41	0.75	41	?	3	480	n/a
Yuneec Power Drive 400	2.11	?	7	n/a	4	70	n/a
Strecker Motoren 990.8	4	?	?	n/a	5	37.5	n/a
AeroConversions	1.63	?	?	?	4	90.8	n/a
SkySpark	1.5	1.75	?	n/a	3	120	n/a
PMG 132	0.4	1.01	?	n/a	34	380.8	n/a
EM42	1.32	?	9.6	n/a	4	116.4	n/a

Table A.3: Motor survey results part 3/3.

Appendix B:

PMBLDC Six-Step Drive Equations

The general form of the solutions for all six commutation periods is:

$$p\begin{bmatrix} i_1 \\ i_2 \end{bmatrix} = \frac{-R_{ph}}{L-M}\begin{bmatrix} i_1 \\ i_2 \end{bmatrix} - \frac{1}{3(L-M)}\begin{bmatrix} -2 & 1 \\ 1 & 1 \end{bmatrix}\begin{bmatrix} (v_{12}-e_{12}) \\ (v_{23}-e_{23}) \end{bmatrix} \tag{B.1}$$

For the first commutation period $30 \le \theta_e < 90$:

$$\begin{aligned} v_{12} &= v_{ab} & v_{23} &= v_{cb} \\ e_{12} &= e_{ab} & e_{23} &= e_{cb} \\ i_1 &= i_a & i_2 &= i_b \end{aligned} \tag{B.2}$$

where θ_e is the rotor angle in electrical degrees,

$v_{ab} = v_a - v_b$ is the voltage between phases A and B,

$e_{ab} = e_a - e_b$ is the difference in back-EMF between phases A and B.

Other subscripted voltages and back-EMFs in a similar manner. The back-EMF for phase A is aligned to be at its maximum from 30 to 150 electrical degrees. The back-EMF for phase B is shifted 120 electrical degrees, while the back-EMF for phase C is phase shifted 240 electrical degrees.

For the second commutation period $90 \le \theta_e < 150$:

$$\begin{aligned} v_{12} &= v_{ac} & v_{23} &= v_{bc} \\ e_{12} &= e_{ac} & e_{23} &= e_{bc} \\ i_1 &= i_a & i_2 &= i_c \end{aligned} \tag{B.3}$$

For the third commutation period $150 \le \theta_e < 210$:

$$\begin{aligned} v_{12} &= v_{bc} & v_{23} &= v_{ac} \\ e_{12} &= e_{bc} & e_{23} &= e_{ac} \\ i_1 &= i_b & i_2 &= i_c \end{aligned} \tag{B.4}$$

For the fourth commutation period $210 \le \theta_e < 270$:

$$\begin{aligned} v_{12} &= v_{ab} & v_{23} &= v_{cb} \\ e_{12} &= e_{ab} & e_{23} &= e_{cb} \\ i_1 &= i_a & i_2 &= i_b \end{aligned} \tag{B.5}$$

For the fifth commutation period $270 \le \theta_e < 330$:

$$\begin{aligned} v_{12} &= v_{ca} & v_{23} &= v_{ba} \\ e_{12} &= e_{ca} & e_{23} &= e_{ba} \\ i_1 &= i_c & i_2 &= i_a \end{aligned} \tag{B.6}$$

For the sixth commutation period $330 \le \theta_e < 30$:

$$\begin{aligned} v_{12} &= v_{bc} & v_{23} &= v_{ac} \\ e_{12} &= e_{bc} & e_{23} &= e_{ac} \\ i_1 &= i_b & i_2 &= i_c \end{aligned} \tag{B.7}$$

Appendix C:

MATLAB Code for the Flight Time Simulations

```
%This matlab program gives estimates of flight times for an electric
%jetpack given a particualar battery type

batt_mass = 10:1:200;                    %kg, set up range of battery masses to plot
batt_energy_density = 108;               %Whr/kg, energy density of the battery
batt_total_energy = (batt_mass.*batt_energy_density).*3600; %J, energy availiable in
the battery
batt_power_density = 2000; %W/kg, power density of battery

frame_mass = 35;      %kg,
other_mass = 15;      %kg,
pilot_mass = 127;     %kg,
motor_mass = 50;      %kg,
drive_mass = 26;      %kg,
system_eff = 0.90;    %

max_thrust_per_fan = 1500;   %N, safety factor, sets a maximum thrust that each fan
can support default 1500
max_total_thrust = max_thrust_per_fan*2;  %N

thrust_req_per_duct = (frame_mass + pilot_mass + motor_mass + drive_mass + batt_mass
+ other_mass).*(9.81/2);
power_req_per_duct = 0.0203.*thrust_req_per_duct.^2 + 23.787.*thrust_req_per_duct -
612.9;
total_power_req = (power_req_per_duct.*2)./system_eff;
flight_time = (batt_total_energy./total_power_req)./60;

max_allowable_batt_mass = (max_total_thrust/9.81) - frame_mass - pilot_mass -
motor_mass - drive_mass;
batt_power_available = batt_mass.*batt_power_density;
for i=1:length(batt_mass)
    if batt_power_available(i) > total_power_req(i)
        break;
    end
end
min_allowable_batt_mass = batt_mass(i);

plot(batt_mass,flight_time)
xlabel('Weight (kg)');
ylabel('Flight Time (minutes)');
title('Estimated Flight Time vs. Weight Used');
axis([0 1.1*batt_mass(length(batt_mass)) 0 5])
line([max_allowable_batt_mass max_allowable_batt_mass],[0
5],'LineStyle',':','Color','red','Marker','<');
line([min_allowable_batt_mass min_allowable_batt_mass],[0
5],'LineStyle',':','Color','red','Marker','>');
```

Appendix D:

MATLAB Code for the PMBLDC Current Simulations

```
function average_torque = simulate_phase_current(line_line_resistance,
line_line_inductance, desired_max_current, number_poles, back_EMF_constant_per_phase,
supply_voltage_factor, speed, advance_angle, do_current_limit)

%MOTOR PARAMETERS
Rp = line_line_resistance/2;          %phase resistance
Lp = line_line_inductance/2;          %equivalent inductance per phase
Imax = desired_max_current;                  %desired maximum current limit
rpm = speed;                                        %mechanical speed in RPM
Np = number_poles;                                  %number of magnet poles
s_mech = 2*pi*(rpm/60);                             %angular velocity
s_elec = s_mech*(Np/2);                             %electrical angular velocity
f_mech = rpm/60;                                    %mechanical frequency
f_elec = f_mech*(Np/2);                             %electrical frequency
Kb = back_EMF_constant_per_phase;     %peak back-EMF per unit of mechanical speed for
a single phase
Vs_min = (s_mech*Kb+Imax*Rp)*2;              %minimum supply voltage needed to
operate motor at low speed
Vs = Vs_min*supply_voltage_factor;    %supply voltage for the inverter
adv_ang = advance_angle;                     %phase advance angle

number_cycles = 5;
time_per_cycle = 1/f_elec;
time_finish = time_per_cycle*number_cycles;
dt = time_per_cycle/10000;

%INITIAL CONDITIONS:
ia = 0; ib = 0; ic = 0;                             %initial phase currents
theta_mr = 0;                                       %initial position in mechanical
radians
time = 0;                                                   %start time

num_points = ceil((time_finish-time)/dt);
num_graph_points = 10000;
take_samples_every = round(num_points/num_graph_points);
if(take_samples_every < 1)
       take_samples_every = 1;
end
counter = take_samples_every;
index = 1;

%SIMULATE MOTOR
while (time<=time_finish)

       %determine commutation position
       theta_ed = (180*theta_mr/pi)*(Np/2);
       while ((theta_ed<0)||(theta_ed>360))
              if (theta_ed<0)
                     theta_ed = theta_ed+360;
              elseif (theta_ed > 360)
                     theta_ed = theta_ed - 360;
              end
       end
       mod_theta = theta_ed + adv_ang;
       if (mod_theta<0)
                     mod_theta = mod_theta+360;
       elseif (mod_theta > 360)
                     mod_theta = mod_theta - 360;
       end


       if(((mod_theta>=0)&&(mod_theta<30))||((mod_theta>=330)&&(mod_theta<=360)))
              mode = 6;
```

```
elseif ((mod_theta>=30)&&(mod_theta<90))
        mode = 1;
elseif ((mod_theta>=90)&&(mod_theta<150))
        mode = 2;
elseif ((mod_theta>=150)&&(mod_theta<210))
        mode = 3;
elseif ((mod_theta>=210)&&(mod_theta<270))
        mode = 4;
elseif ((mod_theta>=270)&&(mod_theta<330))
        mode = 5;
end

%calculate back-emf shape for phase A
if ((theta_ed>=0)&&(theta_ed<30))
        emf_shape_A = theta_ed/30;
elseif ((theta_ed>=30)&&(theta_ed<150))
        emf_shape_A = 1;
elseif ((theta_ed>=150)&&(theta_ed<210))
        emf_shape_A = -theta_ed/30 + 6;
elseif ((theta_ed>=210)&&(theta_ed<330))
        emf_shape_A = -1;
elseif ((theta_ed>=330)&&(theta_ed<360))
        emf_shape_A = theta_ed/30 - 12;
end

%calculate back-emf shape for phase B
if ((theta_ed>=0)&&(theta_ed<90))
        emf_shape_B = -1;
elseif ((theta_ed>=90)&&(theta_ed<150))
        emf_shape_B = theta_ed/30 - 4;
elseif ((theta_ed>=150)&&(theta_ed<270))
        emf_shape_B = 1;
elseif ((theta_ed>=270)&&(theta_ed<330))
        emf_shape_B = -theta_ed/30 + 10;
elseif ((theta_ed>=330)&&(theta_ed<360))
        emf_shape_B =  -1;
end

%calculate back-emf shape for phase C
if ((theta_ed>=0)&&(theta_ed<30))
        emf_shape_C = 1;
elseif ((theta_ed>=30)&&(theta_ed<90))
        emf_shape_C = -theta_ed/30 + 2;
elseif ((theta_ed>=90)&&(theta_ed<210))
        emf_shape_C = -1;
elseif ((theta_ed>=210)&&(theta_ed<270))
        emf_shape_C = theta_ed/30 - 8;
elseif ((theta_ed>=270)&&(theta_ed<360))
        emf_shape_C =  1;
end

%calculate back-emf for all three phases
emf_A = emf_shape_A*Kb*s_mech;
emf_B = emf_shape_B*Kb*s_mech;
emf_C = emf_shape_C*Kb*s_mech;

delta_i = 0;
%solve for current
if(mode == 1)
        Va = Vs/2;      Vb = -Vs/2;
        if(ic > 0)
                Vc = -Vs/2;
        else
                Vc = emf_C;
        end
        Vab = Va - Vb;
        Vcb = Vc - Vb;
        v12 = Vab;
        v23 = Vcb;
        i1 = ia;
        i2 = ib;
        e12 = emf_A - emf_B;
        e23 = emf_C - emf_B;
        delta_i = calc_currents(v12,v23,i1,i2,e12,e23, Rp, Lp);
        i1 = delta_i(1,1)*dt + i1;
        i2 = delta_i(2,1)*dt + i2;
        i3 = -(i1 + i2);
```

```
        ia = i1; ib = i2; ic = i3;
elseif(mode == 2)
        Va = Vs/2;     Vc = -Vs/2;
        if(ib < 0)
                Vb = +Vs/2;
        else
                Vb = emf_B;
        end
        Vac = Va - Vc;
        Vbc = Vb - Vc;
        v12 = Vac;
        v23 = Vbc;
        i1 = ia;
        i2 = ic;
        e12 = emf_A - emf_C;
        e23 = emf_B - emf_C;
        delta_i = calc_currents(v12,v23,i1,i2,e12,e23, Rp, Lp);
        i1 = delta_i(1,1)*dt + i1;
        i2 = delta_i(2,1)*dt + i2;
        i3 = -(i1 + i2);
        ia = i1; ic = i2; ib = i3;
elseif(mode == 3)
        Vb = Vs/2;     Vc = -Vs/2;
        if(ia > 0)
                Va = -Vs/2;
        else
                Va = emf_A;
        end
        Vbc = Vb - Vc;
        Vac = Va - Vc;
        v12 = Vbc;
        v23 = Vac;
        i1 = ib;
        i2 = ic;
        e12 = emf_B - emf_C;
        e23 = emf_A - emf_C;
        delta_i = calc_currents(v12,v23,i1,i2,e12,e23, Rp, Lp);
        i1 = delta_i(1,1)*dt + i1;
        i2 = delta_i(2,1)*dt + i2;
        i3 = -(i1 + i2);
        ib = i1; ic = i2; ia = i3;
elseif(mode == 4)
        Va = -Vs/2;    Vb = Vs/2;
        if(ic < 0)
                Vc = +Vs/2;
        else
                Vc = emf_C;
        end
        Vab = Va - Vb;
        Vcb = Vc - Vb;
        v12 = Vab;
        v23 = Vcb;
        i1 = ia;
        i2 = ib;
        e12 = emf_A - emf_B;
        e23 = emf_C - emf_B;
        delta_i = calc_currents(v12,v23,i1,i2,e12,e23, Rp, Lp);
        i1 = delta_i(1,1)*dt + i1;
        i2 = delta_i(2,1)*dt + i2;
        i3 = -(i1 + i2);
        ia = i1; ib = i2; ic = i3;
elseif(mode == 5)
        Va = -Vs/2;    Vc = Vs/2;
        if(ib > 0)
                Vb = -Vs/2;
        else
                Vb = emf_B;
        end
        Vca = Vc - Va;
        Vba = Vb - Va;
        v12 = Vca;
        v23 = Vba;
        i1 = ic;
        i2 = ia;
        e12 = emf_C - emf_A;
        e23 = emf_B - emf_A;
        delta_i = calc_currents(v12,v23,i1,i2,e12,e23, Rp, Lp);
```

```
                i1 = delta_i(1,1)*dt + i1;
                i2 = delta_i(2,1)*dt + i2;
                i3 = -(i1 + i2);
                ic = i1; ia = i2; ib = i3;
        elseif(mode == 6)
                Vb = -Vs/2;    Vc = Vs/2;
                if(ia < 0)
                        Va = +Vs/2;
                else
                        Va = emf_A;
                end
                Vbc = Vb - Vc;
                Vac = Va - Vc;
                v12 = Vbc;
                v23 = Vac;
                i1 = ib;
                i2 = ic;
                e12 = emf_B - emf_C;
                e23 = emf_A - emf_C;
                delta_i = calc_currents(v12,v23,i1,i2,e12,e23, Rp, Lp);
                i1 = delta_i(1,1)*dt + i1;
                i2 = delta_i(2,1)*dt + i2;
                i3 = -(i1 + i2);
                ib = i1; ic = i2; ia = i3;
        end
        if(do_current_limit == 1)
                if(ia>desired_max_current)
                        ia = desired_max_current;
                end
                if(ib>desired_max_current)
                        ib = desired_max_current;
                end
                if(ic>desired_max_current)
                        ic = desired_max_current;
                end
                if(ia<(-desired_max_current))
                        ia = -desired_max_current;
                end
                if(ib<(-desired_max_current))
                        ib = -desired_max_current;
                end
                if(ic<(-desired_max_current))
                        ic = -desired_max_current;
                end
        end
        torque = (ia*emf_A + ib*emf_B + ic*emf_C)/s_mech;
        time = time + dt;
        theta_mr = theta_mr + s_mech*dt;

        if(counter == take_samples_every)
                data(1,index) = emf_A;
                data(2,index) = emf_B;
                data(3,index) = emf_C;
                data(4,index) = ia;
                data(5,index) = ib;
                data(6,index) = ic;
                data(7,index) = Va;
                data(8,index) = Vb;
                data(9,index) = Vc;
                data(10,index) = time;
                data(11,index) = torque;
                counter = 0;
                index = index + 1;
        end

        counter = counter + 1;
end

points_per_period = round(length(data(10,:))/number_cycles);
num_data_points = length(data(10,:));
average_torque_data = data(11,(num_data_points-points_per_period):num_data_points);
average_torque = sum(average_torque_data) / points_per_period;

function [grads] = calc_currents(v12,v23,i1,i2,e12,e23, Rp, Lp)
grads = (-Rp/Lp)*[i1;i2] + (-1/(3*Lp))*[-2*(v12 - e12)+(v23 - e23) ; (v12 - e12)+(v23
- e23)];
```

Appendix E:

MATLAB Code for the PMBLDC Simulation Program

Main simulation function:

```
function [trv, heat_flux, efficiency, power_density, power_output, mass_total,
faulty_design, phase_emf] = motorsim( …
                draw_figures,...        %draw graphs of motor waveforms, 0=false, 1=true
                rotor_type,...          %0=inside rotor, 1=outside rotor
                driver_type,...         %0=squarewave, 1=sinewave
                magnet_type,...         %see list in calc_magnet_details.m
                core_type,...           %see list in calc_core_loss.m
                max_rotor_flux_den,... %maximum rotor flux density (T)
                max_stator_flux_den,...%maximum stator flux density (T)
                temp_windings,...       %temperature of windings (degrees celcius)
                temp_magnets,...        %temperature of magnets (degrees celcius)
                winding_connect_method,...%0=wye, 1=delta
                coil_table,...          %list of coils [start stop n_turns ; ...]
                slots_between_phases,...%number of slots to rotate to get next phase
winding
                current_density,...     %maximum RMS current density (A)
                fill_factor,...         %fill factor of bare copper in each slot, 0=0%,
1=100%
                number_poles,...        %number of magnet poles
                number_slots,...        %number of stator slots
                thick_magnet,...        %magnet thickness (m)
                k_mag_ang,...           %angular magnet arc fraction, 0=0%, 1=100%
                radius_gap,...          %radius of the airgap (m)
                length_gap,...          %length of airgap (m)
                radius_stator,...%radius of the stator outside(inner rotor) or
inside(outer rotor)
                length_stator,...%stator axial length (m)
                length_shaft,...        %length of the shaft (m)
                radius_shaft,...        %radius of the shaft (m)
                k_tooth_tip,...         %tooth tip arc fraction, 0=0%, 1=100%
                ang_tooth_tip,...       %tooth tip angle inwards (degrees)
                depth_tt,...            %depth of tooth tip (m)
                motor_rpm,...           %speed of motor (RPM)
                k_leakage,...           %define magnet flux leakage factor, 0.0->1.0
                theta_res_elec,...      %resolution of analysis (electrical degrees)
                supply_voltage_factor...%supply voltage as a fraction of the peak
line-line back emf                      )

%MATERIALS
DEFINITIONS=========================================================================
if (magnet_type == 1)
        magnet_name = 'Neo_N3575';
        density_magnet = 7474;
    rem_flux_den_20 = 1.20;
        max_temp_magnet = 180;
        ur = 1.05;
        temp_coeff = -0.11;
    Hknee = -557042;
        rem_flux_den = rem_flux_den_20 + (temp_magnets -
20)*(temp_coeff/100)*rem_flux_den_20;
end

if (core_type == 1)
    core_name = 'SiFe_M-19_0.014in';
    k_stack = 0.96;
        desnity_core = 7650;
        %core loss coefficients  (calculated using loss per kg, f=400->1500 Hz,
Bmax=1.0-1.5T)
        ce = -7.393E-5*max_stator_flux_den^2 + 2.222E-4*max_stator_flux_den -9.803E-5;
    ch = 0.02815;
    nl=0.4514;
```

```
    ml=0.5935;
        if(max_stator_flux_den > 1.5)
                ce = -7.393E-5*1.5^2 + 2.222E-4*1.5 -9.803E-5;
        end
end

if (core_type == 2)
    core_name = 'Cobalt-Iron_Hyperco_50A_0.006in';
    k_stack = 0.895;
        desnity_core = 8000;
        %core loss coefficients  (calculated using loss per kg, f=60->1200Hz,
Bmax=1.0-2.0T)
        ce = -3.899E-6*max_stator_flux_den + 2.2076E-5;
    ch = 0.01420136;
    nl=1.414377319;
    ml=0.1833832469;
        if(max_stator_flux_den > 1.5)
                ce = -3.899E-6*max_stator_flux_den + 2.2076E-5;
        end
end

if (core_type == 3)
    core_name = 'Arnon5_Special_0.005in';
    k_stack = 0.89;
        desnity_core = 7650;
        %core loss coefficients  (calculated using loss per kg, f=400->2000 Hz,
Bmax=1.0-1.5T)
        ce = -1.56E-6*max_stator_flux_den^2 + 3.3283E-6*max_stator_flux_den +9.349E-6;
    ch = 0.02170065;
    nl=0.8192826542;
    ml=0.8403473839;
        if(max_stator_flux_den > 1.5)
                ce = -1.56E-6*1.5^2 + 3.3283E-6*1.5 +9.349E-6;
        end
end

%BEGIN MAIN FUNCTION
HERE================================================================
faulty_design = 0;

%calculate stator and rotor dimensions
if (rotor_type == 0)   %inside rotor
radius_so = radius_stator;
radius_si = radius_gap + length_gap/2;
radius_ro = radius_gap - length_gap/2;
radius_ym = radius_ro - thick_magnet;
radius_mp = radius_ym + thick_magnet/3;
elseif (rotor_type == 1)      %outside rotor
radius_si = radius_stator;
radius_so = radius_gap - length_gap/2;
radius_ri = radius_gap + length_gap/2;
radius_ym = radius_ri + thick_magnet;
radius_mp = radius_ri + thick_magnet/3;
end
angle_tt_mr = (2*pi*k_tooth_tip)/number_slots;
angle_tt_er = angle_tt_mr*(number_poles/2);
angle_tt_md = 180*angle_tt_mr/pi;
angle_tt_ed = 180*angle_tt_er/pi;
angle_tp_mr = (2*pi)/number_slots;
angle_tp_er = angle_tp_mr*(number_poles/2);
angle_tp_md = 180*angle_tp_mr/pi;
angle_tp_ed = 180*angle_tp_er/pi;
ang_tooth_tip_r = ang_tooth_tip*pi/180;

%calculate area per magnet pole and area of airgap per pole
area_mp = (k_mag_ang*2*pi*radius_mp*length_stator)/number_poles;
area_gp = (k_mag_ang*2*pi*radius_gap*length_stator)/number_poles;

%calculate nominal flux and flux densities  (not including slotting effects)
flux_den_airgap = rem_flux_den*((k_leakage*(area_mp /
area_gp))/(1+ur*k_leakage*(area_mp/area_gp)*(length_gap/thick_magnet)));
flux_den_magnet = (flux_den_airgap * area_gp)/(k_leakage*area_mp);
flux_gap = area_gp*flux_den_airgap;
flux_mag = area_mp*flux_den_magnet;

%calculate approximate thickness of stator and rotor yoke
width_sy = flux_gap / (2 * length_stator * max_stator_flux_den * k_stack);
```

```
width_ry = flux_mag / (2 * length_stator * max_rotor_flux_den);

fprintf('Rotor yoke width: %.1f mm\n',width_ry*1000);

width_t =
(flux_den_airgap*length_stator*pi*angle_tt_mr)/(length_stator*max_stator_flux_den*k_s
tack);

%calculate inside or outside rotor radius based on approximate yoke thicknesses
if (rotor_type == 0)
       radius_ri = radius_ym - width_ry;
elseif (rotor_type == 1)
       radius_ro = radius_ym + width_ry;
end

%set up stuff needed for waveforms and analysis
theta_e = 0:theta_res_elec:360;
num_points = length(theta_e);
theta_res_mech = theta_res_elec/(number_poles/2);
time_between_points =
((theta_res_mech*2*pi/360)*radius_gap)/(radius_gap*2*pi*motor_rpm/60);

%calculate the stator tooth flux waveform
tooth_flux = zeros(1,num_points);
tooth_flux = calc_tooth_flux(theta_res_elec,number_poles, k_mag_ang, k_tooth_tip,
flux_den_airgap, angle_tp_ed, radius_gap, length_gap, thick_magnet, ur,
length_stator);
tooth_flux_max = max(tooth_flux);

%calculate the stator yoke flux waveform
yoke_flux = zeros(1,num_points);
yoke_flux = calc_yoke_flux(theta_res_elec, tooth_flux, number_slots, angle_tp_ed);
yoke_flux_max = max(yoke_flux);

%resize stator yoke width and stator tooth width based on accurate flux waveforms
width_sy = yoke_flux_max/(length_stator*max_stator_flux_den*k_stack);fprintf('Stator
yoke width: %.1f mm\n',width_sy*1000);
width_t = tooth_flux_max/(length_stator*max_stator_flux_den*k_stack);fprintf('Tooth
width: %.1f mm\n',width_t*1000);

%calculate the single tooth single turn back-EMF waveform
tooth_emf = -diff(tooth_flux)./time_between_points;
tooth_emf(num_points) = tooth_emf(1);

%calculate the phase back-EMF waveform
phase_emf = zeros(1,num_points);
phase_emf = calc_phase_emf(theta_res_elec, tooth_emf, coil_table, number_slots,
angle_tp_ed);
peak_phase_emf = max(phase_emf);fprintf('Peak phase back-EMF: %.1f
V\n',peak_phase_emf);

%calculate the area per tooth, and the augmented area used for core loss calculations
[tooth_area, augmented_tooth_area, area_per_slot, faulty_design] =
calc_tooth_area(rotor_type, width_t, radius_si, radius_so, width_sy, angle_tt_mr,
ang_tooth_tip_r, faulty_design, number_slots, depth_tt);

%calculate the rms current per phase
total_conductors_per_phase = 2*sum(abs(coil_table(:,3)));
slot_area_per_phase = (number_slots/3)*area_per_slot;
copper_area_per_phase = slot_area_per_phase*fill_factor;
copper_area_per_conductor = copper_area_per_phase/total_conductors_per_phase;
current_per_phase = copper_area_per_conductor*current_density;fprintf('Phase current:
%.1f Arms\n',current_per_phase);

%calculate the phase current waveform and rms phase current
phase_current = calc_phase_current(driver_type, winding_connect_method,
current_per_phase, phase_emf, time_between_points, theta_res_elec, theta_e);
peak_phase_current = max(phase_current);fprintf('Peak phase current: %.1f
A\n',peak_phase_current);

%calculate phase torque waveform
rot_velocity = (motor_rpm/60)*2*pi;
phase_torque = zeros(1,num_points);
phase_torque = phase_current .* phase_emf ./ rot_velocity;

%calculate total torque waveform
rotation_amount = round((num_points-1)/3);
```

```
total_torque = phase_torque(1:num_points-1) + circshift(phase_torque(1:num_points-
1),[0 rotation_amount]) + circshift(phase_torque(1:num_points-1),[0
2*rotation_amount]);
total_torque(num_points) = total_torque(1);
average_torque = mean(total_torque(1:num_points-1));fprintf('Average torque: %.1f
Nm\n',average_torque);

%calculate the average radius of the windings and the thickness of the bundle
if(rotor_type == 0)    %inner rotor
       radius_wind = ((radius_so-width_sy)-radius_si)/2 + radius_si;
elseif(rotor_type == 1) %outer rotor
       radius_wind = radius_so - (radius_so - (radius_si+width_sy))/2;
end

%calculate the resistance per phase and the total length of copper
[phase_resistance, total_length_copper] = calc_phase_resistance(coil_table,
number_slots, rotor_type, radius_wind, length_stator, temp_windings,
copper_area_per_conductor);

%calculate resistive power loss per phase waveform
phase_power_loss = phase_resistance * current_per_phase^2;
total_resistive_loss = 3*phase_power_loss;fprintf('Resistive losses: %.1f
W\n',total_resistive_loss);

%calculate the mass of the motor
density_copper = 8940;
density_steel = 7840;
mass_copper = 3*total_length_copper*copper_area_per_conductor*density_copper;
if(rotor_type == 0)    %inner rotor
       mass_sy = (pi*radius_so^2 - pi*(radius_so-width_sy)^2) * length_stator *
desnity_core * k_stack;
       mass_ry = (pi*radius_ym^2 - pi*radius_ri^2) * length_stator * density_steel;
       mass_magnets = (pi*radius_ro^2 - pi*(radius_ro-
thick_magnet)^2)*k_mag_ang*density_magnet*length_stator;
elseif(rotor_type == 1)       %outer rotor
       mass_sy = (pi*(radius_si+width_sy)^2 - pi*radius_si^2) * length_stator *
desnity_core * k_stack;
       mass_ry = (pi*radius_ro^2 - pi*radius_ym^2) * length_stator * density_steel;

       mass_magnets = (pi*radius_ym^2 - pi*(radius_ym-
thick_magnet)^2)*k_mag_ang*density_magnet*length_stator;
end
mass_teeth = tooth_area * length_stator * desnity_core * k_stack * number_slots;
mass_teeth_aug = augmented_tooth_area * length_stator * desnity_core * k_stack *
number_slots;
%mass_shaft = length_shaft * pi*radius_shaft^2 * density_steel;
mass_shaft = 0;
mass_total = mass_copper + mass_sy + mass_ry + mass_teeth + mass_magnets +
mass_shaft;

%calculate the core losses
tooth_flux_density = tooth_flux./(length_stator*k_stack*width_t);
yoke_flux_density = yoke_flux./(length_stator*k_stack*width_sy);
ms_tooth_flux_diff = mean((diff(tooth_flux_density)./time_between_points).^2);
ms_yoke_flux_diff = mean((diff(yoke_flux_density)./time_between_points).^2);
freq_e = (number_poles/2)*motor_rpm/60;
loss_hyst_teeth =
mass_teeth_aug*ch*freq_e*max_stator_flux_den^(nl+ml*max_stator_flux_den);
loss_eddy_teeth = mass_teeth_aug*(ce/(2*pi^2))*(ms_tooth_flux_diff);
loss_hyst_yoke =  mass_sy*ch*freq_e*max_stator_flux_den^(nl+ml*max_stator_flux_den);
loss_eddy_yoke = mass_sy*(ce/(2*pi^2))*(ms_yoke_flux_diff);
loss_core = loss_hyst_teeth + loss_eddy_teeth + loss_hyst_yoke +
loss_eddy_yoke;fprintf('Core losses: %.1f W\n',loss_core);

%calculate the self inductance per phase, mutual inductance, and the line-line
inductance and resistance
if (rotor_type == 0) %inner rotor
       slot_depth = (radius_so - width_sy) - (radius_si+depth_tt);
       slot_width = ((2*pi*radius_wind) - number_slots*width_t)/number_slots;
       tooth_opening_width = ((2*pi*radius_si)*(1-k_tooth_tip))/number_slots;
elseif (rotor_type == 1) %outer rotor
       slot_depth = (radius_so-depth_tt) - (radius_si + width_sy);
       slot_width = ((2*pi*radius_wind) - number_slots*width_t)/number_slots;
       tooth_opening_width = ((2*pi*radius_so)*(1-k_tooth_tip))/number_slots;
end
[phase_inductance, phase_inductance_airgap, phase_inductance_endturns,
phase_inductance_leakage, mutual_inductance] = calc_phase_inductance(coil_table,
```

```
slots_between_phases, number_slots, number_poles, radius_gap, length_stator,
radius_mp, ur, length_gap, thick_magnet, radius_wind, area_per_slot, k_tooth_tip,
slot_depth, slot_width, depth_tt, tooth_opening_width);
if (winding_connect_method == 0)              %wye
       line_line_inductance = 2*phase_inductance - 2*mutual_inductance;fprintf('Line-
line inductance: %.2f mH\n',line_line_inductance*1000);
       line_line_resistance = 2*phase_resistance;fprintf('Line-line resistance: %.1f
mOhm\n',line_line_resistance*1000);
elseif (winding_connect_method == 1)  %delta (to be checked)
       eq_ind_per_phase = phase_inductance - mutual_inductance;
       line_line_inductance = 1/(1/eq_ind_per_phase + 1/(2*eq_ind_per_phase));
       line_line_resistance = 1/(1/phase_resistance + 1/(2*phase_resistance));
end

%calculate the time constant
LR_time_constant = line_line_inductance / line_line_resistance;
time_per_electrical_cycle = (1/(motor_rpm/60))/(number_poles/2);
fraction_of_cycle = LR_time_constant/time_per_electrical_cycle;fprintf('Time-constant
fraction: %.1f\n',fraction_of_cycle);
supply_voltage = supply_voltage_factor*peak_phase_emf*2;
max_time_constant = calc_max_time_constant(line_line_inductance,
line_line_resistance, motor_rpm, number_poles, peak_phase_current, peak_phase_emf,
supply_voltage);
if(LR_time_constant > max_time_constant)
       faulty_design = 2;
end

%calculate the power output
power_output = average_torque * rot_velocity;

%calculate the power density
power_density = power_output / mass_total;

%calculate the efficiency
efficiency = power_output / (power_output+loss_core+phase_power_loss*3);

%calculate the heat flux per unit stator surface area
if (rotor_type == 0)
       surface_area = 2*pi*radius_so*length_stator;
elseif (rotor_type == 1)
       surface_area = 2*pi*radius_si*length_stator;
end
heat_flux = (loss_core + phase_power_loss*3)/surface_area;

%calculate the torque-per-unit-rotor-volume (TRV)
trv = average_torque / (pi*radius_gap^2*length_stator);

%determine whether demagnetisation is occurring
[Hmag] = calc_magnet_operating_point(coil_table, number_slots, radius_gap,
length_stator, radius_mp, ur, length_gap, thick_magnet, radius_wind, area_per_slot,
peak_phase_current, k_tooth_tip, tooth_flux_max, k_leakage, rem_flux_den);
if(Hmag < Hknee)
       faulty_design = 3;
end

%check for a faulty design
if (rotor_type == 0) %inner rotor
       if(            (radius_so > (radius_so-width_sy))&&...
                      ((radius_so-width_sy) > radius_si)&&...
                      (radius_si > radius_gap)&&...
                      (radius_gap > radius_ro)&&...
                      (radius_ro > radius_ym)&&...
                      (radius_ym > radius_ri)&&...
                      (radius_ri > radius_shaft)           )
              faulty_design = faulty_design;
       else
              faulty_design = 4;
       end
elseif (rotor_type == 1) %outer rotor
       if(            (radius_ro > radius_ym)&&...
                      (radius_ym > radius_ri)&&...
                      (radius_ri > radius_gap)&&...
                      (radius_gap > radius_so)&&...
                      (radius_so > (radius_si+width_sy))&&...
                      ((radius_si+width_sy) > radius_si)&&...
                      (radius_si > radius_shaft)           )
              faulty_design = faulty_design;
```

```
        else
                faulty_design = 5;
        end
end

if((phase_power_loss>0)&&(loss_core>0))
        faulty_design = faulty_design;
else
        faulty_design = 6;
end



if (draw_figures == 1)
        figure(1);
        plot(theta_e, tooth_flux);
        title('Tooth flux waveform');
        figure(2);
        plot(theta_e, yoke_flux);
        title('Yoke flux waveform');
        figure(3);
        plot(theta_e, tooth_emf);
        title('Tooth EMF waveform');
        figure(4);
        hold on;
        plot(theta_e, phase_emf);
        title('Phase EMF waveform');
    plot(theta_e, phase_current);
        figure(5);
        plot(theta_e, phase_current);
        title('Phase current waveform');
        figure(6);
        plot(theta_e, phase_torque);
        axis([0 400 0 1.2*max(phase_torque)]);
        title('Phase torque waveform');
    figure(7);
        plot(theta_e, total_torque);
        axis([0 400 0 1.2*max(total_torque)]);
        title('Total torque waveform');
        figure('Position',[200,200,400,300]);
        plot(theta_e, phase_emf);
        title('Phase Back-EMF Waveform');
        ylabel('Back-EMF (V)')
        xlabel('Rotor Angle (elec. deg)')
        axis([0, 360, -peak_phase_emf*1.1, peak_phase_emf*1.1])

        figure('Position',[200,550,400,300]);
        draw_motor(rotor_type, width_t, width_sy, width_ry, radius_si, radius_so,
radius_ri, radius_ro, radius_ym, angle_tt_mr, ang_tooth_tip_r, number_slots,
number_poles, k_mag_ang, depth_tt);
        if (rotor_type == 0)
                axis([-radius_so radius_so -radius_so radius_so]);
        elseif (rotor_type == 1)
                axis([-radius_ro radius_ro -radius_ro radius_ro]);
        end
        axis square;
        set(gca,'YTick',[-.1 -.05 0 .05 .1])
        set(gca,'XTick',[-.1 -.05 0 .05 .1])
end

generate_autocad(rotor_type, width_t, width_sy, width_ry, radius_si, radius_so,
radius_ri, radius_ro, radius_ym, angle_tt_mr, ang_tooth_tip_r, number_slots,
number_poles, k_mag_ang, depth_tt);
```

Tooth flux calculation function:

```
function [tooth_flux] = calc_tooth_flux(theta_res_elec, number_poles, k_mag_ang,
k_tooth_tip, flux_den_airgap, angle_tp_ed, radius_gap, length_gap, thick_magnet, ur,
length_stator)

num_cycles = 3;%number of electrical cycles needed to perform alanysis  |^|_|^|_|^|_i
theta_res_mech = theta_res_elec/(number_poles/2);    %resolution of analysis in
mechanical degrees
num_points = length(0:theta_res_elec:360*num_cycles);%number of points in the
analysis
Bg_wave = zeros(1,num_points);        %array for the flux density waveform (T)
tooth_flux = zeros(1,num_points);     %store the tooth flux waveform (Wb)
tooth_pos = zeros(1,num_points);      %stores tooth tip position waveform
Ksl = zeros(1,num_points);            %stores slot correction factor
angle_tt_ed = angle_tp_ed * k_tooth_tip;

%transitions points in airgap flux waveform in electrical degrees
pt1 = 0;
pt2 = (1-k_mag_ang)*180/2;
pt3 = 180 - (1-k_mag_ang)*180/2;
pt4 = 180 + (1-k_mag_ang)*180/2;
pt5 = 360 - (1-k_mag_ang)*180/2;
pt6 = 360;

%construct squarewave like flux waveform for first elec cycle
for ang=0:theta_res_elec:360
        index = round(ang/theta_res_elec+1);
        if ((ang>=pt1)&&(ang<pt2))
                Bg_wave(index) = 0;
        elseif ((ang>=pt2)&&(ang<pt3))
                Bg_wave(index) = flux_den_airgap;
        elseif ((ang>=pt3)&&(ang<pt4))
                Bg_wave(index) = 0;
        elseif ((ang>=pt4)&&(ang<pt5))
                Bg_wave(index) = -flux_den_airgap;
        elseif ((ang>=pt5)&&(ang<pt6))
                Bg_wave(index) = 0;
        end
end

%copy airgap flux waveform over the remaining electrical cycles
num_elements_per_cycle = (num_points-1)/num_cycles;
Bg_wave(num_points) = '';
rotation_amount = num_elements_per_cycle;
Bg_wave = Bg_wave + circshift(Bg_wave,[0 rotation_amount]) + circshift(Bg_wave,[0
2*rotation_amount]);
Bg_wave(num_points) = Bg_wave(1);

%go through every angle of tooth position over one electrical cycle (starting at 270
degrees)
for tooth_angle=270:theta_res_elec:630

        %construct tooth pitch position waveform for the particular tooth_angle
        for ang=0:theta_res_elec:360*num_cycles
                if ((ang>=(tooth_angle-
angle_tp_ed/2))&&(ang<(tooth_angle+angle_tp_ed/2)))
                        tooth_pos(round(ang/theta_res_elec+1))=1;
                else
                        tooth_pos(round(ang/theta_res_elec+1))=0;
                end
        end

        %construct normalized airgap length over tooth
        ang_tooth_start = tooth_angle - k_tooth_tip*angle_tp_ed/2;
        ang_tooth_end = tooth_angle + k_tooth_tip*angle_tp_ed/2;
        ang_tooth_middle = tooth_angle;
        ang_start = tooth_angle - angle_tp_ed/2;
        ang_end = tooth_angle + angle_tp_ed/2;
        for ang=0:theta_res_elec:360*num_cycles
                ind = round(ang/theta_res_elec+1);
                if ((ang>=ang_start)&&(ang<ang_tooth_start))
                        Ksl(ind) = 1 - (pi*radius_gap*(pi/180)*((ang-
tooth_angle)+angle_tt_ed/2))/(number_poles*length_gap);
                elseif ((ang>=ang_tooth_start)&&(ang<ang_tooth_end))
```

```
                Ksl(ind)=1;
            elseif ((ang>=ang_tooth_end)&&(ang<ang_end))
                Ksl(ind) = 1 + (pi*radius_gap*(pi/180)*((ang-tooth_angle)-
angle_tt_ed/2))/(number_poles*length_gap);
            else
                Ksl(round(ang/theta_res_elec+1))=0;
            end
        end

        %construct slot correction factor over tooth
        Ksl =
(1+thick_magnet./(length_gap.*ur))./(Ksl+thick_magnet./(length_gap.*ur));

        %calculate waveform of flux density entering tooth
        Bg_tooth = tooth_pos .* Ksl .* Bg_wave;
        tooth_flux(round(tooth_angle/theta_res_elec+1)) =
(length_stator*2*pi*theta_res_mech*radius_gap/360)*trapz(Bg_tooth);
end

%truncate waveforms so it only contains a single cycle of the tooth flux that covers
360 degrees
first_element = 270/theta_res_elec + 1;
last_element = 630/theta_res_elec + 1;
tooth_flux = tooth_flux(first_element:last_element);
```

Phase back-EMF calculation function:

```
function [phase_emf] = calc_phase_emf(theta_res_elec, tooth_emf, coil_table,
number_slots, angle_tp_ed)

num_points = length(tooth_emf);
phase_emf = zeros(1,num_points);
num_coils = size(coil_table,1);
phase_emf(num_points) = '';
for k=1:num_coils
        slot_start = coil_table(k,1);
        slot_finish = coil_table(k,2);
        num_turns = coil_table(k,3);
        if (slot_finish > slot_start)
                coil_span = slot_finish - slot_start;
        elseif (slot_finish < slot_start)
                coil_span = (number_slots-slot_start) + slot_finish;
        end
        for i=1:coil_span
                current_tooth = slot_start + (i-1);
                angle_of_current_tooth = (current_tooth-1)*angle_tp_ed;
                rotation_amount = round(angle_of_current_tooth/theta_res_elec);
                phase_emf = phase_emf + num_turns.*circshift(tooth_emf(1:num_points-
1),[0 rotation_amount]);
        end
end
phase_emf(num_points) = phase_emf(1);
```

Tooth area calculation function:

```
function [area_tooth, area_tooth_augmented, area_per_slot, faulty_design_out] =
calc_tooth_area(rotor_type, width_t, radius_si, radius_so, width_sy, angle_tt_mr,
ang_tooth_tip_r, faulty_design, number_slots, depth_tt)

if (rotor_type == 0)   %inner rotor

        x2 = 0;
        y2 = radius_si;
        x1 = y2*sin(angle_tt_mr/2);
        y1=y2*cos(angle_tt_mr/2);
        x3 = (y2+depth_tt)*sin(angle_tt_mr/2);
        y3=(y2+depth_tt)*cos(angle_tt_mr/2);
                ang_1 = pi - (angle_tt_mr/2) - ang_tooth_tip_r;
                ang_2 = (pi/2) - ang_1;
        x4 = width_t/2;
        y4 = (x3 - x4)*tan(ang_2) + y3;
        x6 = 0;
        y6 = radius_so - width_sy;
        x5 = width_t/2;
```

```
        y5 = sqrt(y6^2-x5^2);
        x7 = 0;                                                         y7 =
        radius_so;

        area1 = x1*sqrt(y2^2-x1^2)/2 + y2^2*asin(x1/y2)/2;
        area2 = (x3-x1)*(y1+y3)/2;
        area3 = (x3*sqrt(y6^2-x3^2)/2 + y6^2*asin(x3/y6)/2) - (x5*sqrt(y6^2-x5^2)/2 +
        y6^2*asin(x5/y6)/2);
        area4 = (x3-x4)*(y4+y3)/2;
        area5 = area3 - area4;
        area6 = x3*sqrt(y6^2-x3^2)/2 + y6^2*asin(x3/y6)/2;
        area7 = 0.5*(x3^2+y3^2)*(angle_tt_mr/2);
        area8 = 0.5*(x1^2+y1^2)*(angle_tt_mr/2);
        area9 = area7 - area8;

        area_tooth = (area6 - area1 - area2 - area5)*2;

        area10 = (area_tooth - 2*area9)/2;
        area11 = x5*sqrt(y6^2-x5^2)/2 + y6^2*asin(x5/y6)/2;
        area12 = (x5-0)*(y7+y5)/2;
        area13 = area12 - area11;

        area_tooth_augmented = area_tooth + 2*area13;

        area_tooth_wind = area10*2;

        area_per_slot = (pi*y6^2 - pi*(x3^2+y3^2)-
        number_slots*area_tooth_wind)/number_slots;

        if((x4>0)&&(x1>x4)&&(x3>x1)&&(y1<y2)&&(y1<y3)&&(y3<y4)&&(y4<y5)&&(y5<y6)&&(y6<
        y7)&&(area_tooth>=0)&&(area_tooth_augmented>=0))
                faulty_design_out = faulty_design;
        else
                faulty_design_out = 1;
        end
        elseif (rotor_type == 1)      %outer rotor
        x1 = 0;
        y1 = radius_so;
        x2 = y1*sin(angle_tt_mr/2);
        y2=y1*cos(angle_tt_mr/2);
        x3 = (y1-depth_tt)*sin(angle_tt_mr/2);
        y3=(y1-depth_tt)*cos(angle_tt_mr/2);
        ang_1 = ang_tooth_tip_r - (pi/2) - (angle_tt_mr/2);
        x4 = width_t/2;
        y4 = y3 - (x3 - x4)*tan(ang_1);
        x5 = 0;
        y5 = radius_si + width_sy;
        x6 = width_t/2;
        y6 = sqrt(y5^2-x6^2);
        x7 = 0;
        y7 = radius_si;

        area1 = x4*sqrt(y1^2-x4^2)/2 + y1^2*asin(x4/y1)/2;
        area2 = (x4-0)*(y6+y7)/2;
        area3 = x4*sqrt(y5^2-x4^2)/2 + y5^2*asin(x4/y5)/2;
        area4 = area1 - area3;
        area5 = (x2*sqrt(y1^2-x2^2)/2 + y1^2*asin(x2/y1)/2) - area1;
        area6 = (x3-x4)*(y3+y4)/2;
        area7 = (x2-x3)*(y2+y3)/2;
        area8 = area5 - area6 - area7;

        area_tooth = (area8 + area4)*2;
        area_tooth_augmented = (area8 + area1 - area2)*2;

        area9 = 0.5*(x2^2+y2^2)*(angle_tt_mr/2);
        area10 = 0.5*(x3^2+y3^2)*(angle_tt_mr/2);
        area11 = area9 - area10;

        area_tooth_wind = ((area_tooth/2) - area11)*2;
        area_per_slot = (pi*(x3^2+y3^2) - pi*y5^2 -
        number_slots*area_tooth_wind)/number_slots;

        if((x4>0)&&(x3>x4)&&(x2>x3)&&(y1>y2)&&(y2>y3)&&(y2>y4)&&(y4>y5)&&(y5>y6)&&(y6>
y7)&&(area8>=0)&&(area4>=0)&&(area_tooth>=0)&&(area_tooth_augmented>=0))
                faulty_design_out = faulty_design;
        else
                faulty_design_out = 1; end;end;
```

Phase current calculation function:

```
function [phase_current] = calc_phase_current(driver_type, winding_connect_method,
current_per_phase, phase_emf, time_between_points, theta_res_elec, theta_e)

num_points = length(phase_emf);

%calculate the angle and index with peak fundamental phase back-EMF
T = (num_points-1)*time_between_points;
f0 = 1 / T;
x = 0:time_between_points:T-time_between_points;
Y = 2/(num_points-1) * fft(phase_emf(1:num_points - 1));
fund_emf = real(Y(1+1))*cos(1*2*pi*f0*x) + imag(Y(1+1))*sin(1*2*pi*f0*x);
fund_emf(num_points) = fund_emf(1);
fund_emf = fliplr(fund_emf);
[val,index_peak] = max(fund_emf);
angle_peak = theta_e(index_peak);

%calculate the peak current per phase
if (driver_type == 0) %squarewave
        if (winding_connect_method == 0)
                peak_curr_per_phase = current_per_phase/sqrt(2/3);
        elseif (winding_connect_method == 1)
                peak_curr_per_phase = current_per_phase/sqrt(0.5);
        end
elseif (driver_type == 1) %sinusoidal
        if (winding_connect_method == 0)
                peak_curr_per_phase = current_per_phase*sqrt(2);
        elseif (winding_connect_method == 1)
                peak_curr_per_phase = current_per_phase*sqrt(2);
        end
end

%create phase current waveform
phase_current_proto = zeros(1,num_points-1);
if (driver_type == 0)
        if(winding_connect_method == 0)
                pt1 = 0;
                pt2 = 1*360/12;
                pt3 = 5*360/12;
                pt4 = 7*360/12;
                pt5 = 11*360/12;
                pt6 = 12*360/12;
                for ang=0:theta_res_elec:360-theta_res_elec
                        index = round(ang/theta_res_elec+1);
                        if ((ang>=pt1)&&(ang<pt2))
                                phase_current_proto(index) = 0;
                        elseif ((ang>=pt2)&&(ang<pt3))
                                phase_current_proto(index) = peak_curr_per_phase;
                        elseif ((ang>=pt3)&&(ang<pt4))
                                phase_current_proto(index) = 0;
                        elseif ((ang>=pt4)&&(ang<pt5))
                                phase_current_proto(index) = -peak_curr_per_phase;
                        elseif ((ang>=pt5)&&(ang<=pt6))
                                phase_current_proto(index) = 0;
                        end
                end
        elseif (winding_connect_method == 1)
                pt1 = 0;
                pt2 = 1*360/6;
                pt3 = 2*360/6;
                pt4 = 3*360/6;
                pt5 = 4*360/6;
                pt6 = 5*360/6;
                pt7 = 6*360/6;
                for ang=0:theta_res_elec:360-theta_res_elec
                        index = round(ang/theta_res_elec+1);
                        if ((ang>=pt1)&&(ang<pt2))
                                phase_current_proto(index) = peak_curr_per_phase/2;
                        elseif ((ang>=pt2)&&(ang<pt3))
                                phase_current_proto(index) = peak_curr_per_phase;
                        elseif ((ang>=pt3)&&(ang<pt4))
                                phase_current_proto(index) = peak_curr_per_phase/2;
                        elseif ((ang>=pt4)&&(ang<pt5))
                                phase_current_proto(index) = -peak_curr_per_phase/2;
                        elseif ((ang>=pt5)&&(ang<pt6))
```

```
                        phase_current_proto(index) = -peak_curr_per_phase;
                elseif ((ang>=pt6)&&(ang<=pt7))
                        phase_current_proto(index) = -peak_curr_per_phase/2;
                end
        end
    end
elseif (driver_type == 1)     %sinusiodal
    phase_current_proto = peak_curr_per_phase*sin(pi.*[0:theta_res_elec:360-
theta_res_elec]./180);
end
angle_difference = angle_peak - 90;
rotation_amount = round(angle_difference/theta_res_elec);
phase_current = circshift(phase_current_proto,[0 rotation_amount]);
phase_current(num_points) = phase_current(1);
```

Phase resistance calculation function:

```
function [phase_resistance, total_length_copper] = calc_phase_resistance(coil_table,
number_slots, rotor_type, radius_wind, length_stator, temp_windings,
copper_area_per_conductor)

a_cu = 0.004041;       %temperature coefficient of copper (per deg C)
p_cu = 1.712*10^(-8);  %resistivity of copper at 25 deg C (ohm.m)
total_length_copper = 0;
total_number_turns = 0;

num_coils = size(coil_table,1);
for k=1:num_coils
    slot_start = coil_table(k,1);
    slot_finish = coil_table(k,2);
    num_turns = abs(coil_table(k,3));

    %find coil span
    if (slot_finish > slot_start)
        coil_span = slot_finish - slot_start;
    elseif (slot_finish < slot_start)
        coil_span = (number_slots-slot_start) + slot_finish;
    end

    angle_coil_span = ((2*pi)/number_slots) * coil_span;
    length_coil_per_end = pi*(angle_coil_span*radius_wind)/2;
    length_coil_axial = length_stator;
    length_per_turn = length_coil_per_end*2 + length_coil_axial*2;

    total_length_copper = total_length_copper + length_per_turn*num_turns;
    total_number_turns = total_number_turns + num_turns;
end

%calculate the resistance per phase
phase_resistance_rt = p_cu * total_length_copper / (copper_area_per_conductor);
phase_resistance = phase_resistance_rt*(1+a_cu*(temp_windings-25));
```

Inductance calculation function:

```
function [phase_inductance, phase_inductance_airgap, phase_inductance_endturns,
phase_inductance_leakage, mutual_inductance] = calc_phase_inductance(coil_table,
slots_between_phases, number_slots, number_poles, radius_gap, length_stator,
radius_mp, ur, length_gap, thick_magnet, radius_wind, area_per_slot, k_tooth_tip,
slot_depth, slot_width, depth_tt, tooth_opening_width)

%calculate airgap inductance
base_number_turns_per_coil = min(abs(coil_table(:,3)));
sk_array = zeros(1,number_slots);
num_coils = size(coil_table,1);
for k=1:num_coils
    slot_start = coil_table(k,1);
    slot_finish = coil_table(k,2);
    num_turns = coil_table(k,3);
    if (slot_finish > slot_start)
        coil_span = slot_finish - slot_start;
    elseif (slot_finish < slot_start)
        coil_span = (number_slots-slot_start) + slot_finish;
    end
    for i=1:coil_span
```

```
                current_tooth = slot_start + (i-1);
                if (current_tooth > number_slots)
                        current_tooth = current_tooth - number_slots;
                end
                sk_array(current_tooth) = sk_array(current_tooth) +
num_turns/base_number_turns_per_coil;
        end
end

phase_inductance_airgap = 0;
sum1 = 0;
for m = 1:number_slots
        sum1 = sum1 + sk_array(m);
end
sum2 = 0;
for k = 1:number_slots
        sum2 = sum2 + sign(sk_array(k))*(sk_array(k)-(1/number_slots)*sum1);
end
u0 = 4*pi*10^(-7);
area_gap_per_tooth = (2*pi*radius_gap*length_stator*k_tooth_tip)/number_slots;
%area_gap_per_tooth = (2*pi*radius_gap*length_stator)/number_slots
area_magnet_per_tooth = (2*pi*radius_mp*length_stator*k_tooth_tip)/number_slots;
%area_magnet_per_tooth = (2*pi*radius_mp*length_stator)/number_slots
reluctance_per_tooth = length_gap/(u0*area_gap_per_tooth) +
thick_magnet/(ur*u0*area_magnet_per_tooth);
phase_inductance_airgap = sum2*(base_number_turns_per_coil^2 / reluctance_per_tooth)


%calculate endturn inductance
phase_inductance_endturns = 0;
for k=1:num_coils
        slot_start = coil_table(k,1);
        slot_finish = coil_table(k,2);
        num_turns = coil_table(k,3);
        if (slot_finish > slot_start)
                coil_span = slot_finish - slot_start;
        elseif (slot_finish < slot_start)
                coil_span = (number_slots-slot_start) + slot_finish;
        end
        coil_span_rad = (2*pi/number_slots)*coil_span;
        diameter_of_coil_arc = coil_span_rad*radius_wind;
        GMD = 0.447*sqrt(area_per_slot);
        phase_inductance_endturns = phase_inductance_endturns +
(u0*num_turns^2*diameter_of_coil_arc/2)*log((4*diameter_of_coil_arc)/GMD -2);

end
phase_inductance_endturns

%calculate the slot leakage self inductance
conductors_per_slot_array = zeros(1,number_slots);  %stores the number of conductors
in each slot for a single phase

for k=1:num_coils
        %determine number of conductors in each slot for a single phase
        slot_start = coil_table(k,1);
        slot_finish = coil_table(k,2);
        num_turns = coil_table(k,3);
        conductors_per_slot_array(slot_start) = conductors_per_slot_array(slot_start)
+ abs(num_turns);
        conductors_per_slot_array(slot_finish) =
conductors_per_slot_array(slot_finish) + abs(num_turns);
end

phase_inductance_leakage = 0;
slot_permeance = slot_depth/(3*slot_width) + depth_tt/tooth_opening_width;
for j=1:number_slots
        phase_inductance_leakage = phase_inductance_leakage +
u0*length_stator*conductors_per_slot_array(j)^2*slot_permeance;
end
phase_inductance_leakage
phase_inductance = phase_inductance_airgap + phase_inductance_endturns +
phase_inductance_leakage


%calculate mutual inductance
%number_slots_per_pole = number_slots
original_coils = sk_array*base_number_turns_per_coil;
```

```
rotation_amount = slots_between_phases;
new_coils = circshift(original_coils,[0 rotation_amount]);

%assume 1AMP excitation for flux generation
center_node_mmf = (-base_number_turns_per_coil/number_slots)*sum(sk_array);
original_tooth_flux =
(center_node_mmf+sk_array.*base_number_turns_per_coil)./reluctance_per_tooth;
flux_linked = 0;
for i=1:number_slots
        flux_linked = flux_linked + original_tooth_flux(i)*new_coils(i);
end
mutual_inductance = flux_linked
```

Maximum allowable time constant calculation function:

```
function [max_time_constant] = calc_max_time_constant(line_line_inductance,
line_line_resistance, motor_rpm, number_poles, peak_phase_current, peak_phase_emf,
supply_voltage)

LR_time_constant = line_line_inductance / line_line_resistance;
time_per_electrical_cycle = (1/(motor_rpm/60))/(number_poles/2);
fraction_of_cycle = LR_time_constant/time_per_electrical_cycle;

minimum_SS_voltage_over_resistance = peak_phase_current*line_line_resistance;
actual_SS_voltage_over_resistance = supply_voltage - 2*peak_phase_emf;
fractional_increase = actual_SS_voltage_over_resistance /
minimum_SS_voltage_over_resistance;

if (fractional_increase <= 12)
        allowable_time_constant_fraction = 0.2838*fractional_increase - 0.2933;
elseif (fractional_increase > 12)
        allowable_time_constant_fraction = 0.097*fractional_increase + 1.879;
end

max_time_constant = allowable_time_constant_fraction * time_per_electrical_cycle;
```

Magnet demagnetisation check function:

```
function [Hmag] = calc_magnet_operating_point(coil_table_A, number_slots, radius_gap,
length_stator, radius_mp, ur, length_gap, thick_magnet, radius_wind, area_per_slot,
peak_phase_current, k_tooth_tip, tooth_flux_max, k_leakage, rem_flux_den)

%calculate combined coil table
num_coils_per_phase = size(coil_table_A,1);
num_coils_total = num_coils_per_phase*2;
base_number_turns_per_coil = min(abs(coil_table_A(:,3)));
sk_array = zeros(1,number_slots);

rotation_amount = number_slots / 3;
for j=1:num_coils_per_phase
        coil_table_B(j,1) = coil_table_A(j,1) + rotation_amount;
        coil_table_B(j,2) = coil_table_A(j,2) + rotation_amount;
        if(coil_table_B(j,1)>number_slots)
                coil_table_B(j,1) = (coil_table_B(j,1)-number_slots);
        end
        if(coil_table_B(j,2)>number_slots)
                coil_table_B(j,2) = (coil_table_B(j,2)-number_slots);
        end
        coil_table_B(j,3) = -coil_table_A(j,3);
end
coil_table_combined = [coil_table_A;coil_table_B];

%calculate sk array for single tooth equivalent windings of combined coil table
for k=1:num_coils_total
        slot_start = coil_table_combined(k,1);
        slot_finish = coil_table_combined(k,2);
        num_turns = coil_table_combined(k,3);
        if (slot_finish > slot_start)
                coil_span = slot_finish - slot_start;
        elseif (slot_finish < slot_start)
                coil_span = (number_slots-slot_start) + slot_finish;
```

```
        end
        for i=1:coil_span
                current_tooth = slot_start + (i-1);
                if (current_tooth > number_slots)
                        current_tooth = current_tooth - number_slots;
                end
                sk_array(current_tooth) = sk_array(current_tooth) +
num_turns/base_number_turns_per_coil;
        end
end
u0 = 4*pi*10^(-7);
area_gap_per_tooth = (2*pi*radius_gap*length_stator*k_tooth_tip)/number_slots;
area_magnet_per_tooth = (2*pi*radius_mp*length_stator*k_tooth_tip)/number_slots;
reluctance_per_tooth = length_gap/(u0*area_gap_per_tooth) +
thick_magnet/(ur*u0*area_magnet_per_tooth);


%calculate tooth fluxes due to windings alone
center_node_mmf = (-
base_number_turns_per_coil*peak_phase_current/number_slots)*sum(sk_array);
tooth_flux_for_each_tooth =
(center_node_mmf+sk_array.*base_number_turns_per_coil.*peak_phase_current)./reluctanc
e_per_tooth;

%find maximum flux in airgap per tooth due to windings alone
max_flux_wind_per_tooth = max(abs(tooth_flux_for_each_tooth));

%determine flux from magnet per tooth due to magnet acting alone in the
max_flux_mag_per_tooth = tooth_flux_max/k_leakage;
resultant_flux = max_flux_mag_per_tooth-max_flux_wind_per_tooth;

%determine operating point of magnet under maximum demagnetisation
Hmag = (resultant_flux -
rem_flux_den*area_magnet_per_tooth)/(ur*u0*area_magnet_per_tooth);
```

Draw the motor function:

```
function draw_motor(rotor_type, width_t, width_sy, width_ry, radius_si, radius_so,
radius_ri, radius_ro, radius_ym, angle_tt_mr, ang_tooth_tip_r, number_slots,
number_poles, k_mag_ang, depth_tt)
if (rotor_type == 0)   %inner rotor

        x2 = 0;                                                 y2 = radius_si;
        x1 = y2*sin(angle_tt_mr/2);                             y1=y2*cos(angle_tt_mr/2);
        x3 = (y2+depth_tt)*sin(angle_tt_mr/2);
        y3=(y2+depth_tt)*cos(angle_tt_mr/2);
                ang_1 = pi - (angle_tt_mr/2) - ang_tooth_tip_r;
                ang_2 = (pi/2) - ang_1;
        x4 = width_t/2;                                         y4 = (x3 - x4)*tan(ang_2)
        + y3;
        x6 = 0;                                                 y6 = radius_so -
        width_sy;
        x5 = width_t/2;                                         y5 = sqrt(y6^2-x5^2);
        x7 = 0;                                                 y7 = radius_so;

        %draw the outside stator radius
        hold off;
        number_points = 100;
        resolution = 2*pi/number_points;
        theta = 0:resolution:2*pi;
        circ_x = radius_so*cos(theta);
        circ_y = radius_so*sin(theta);
        plot(circ_x,circ_y);
        hold on;

        %form a prototype tooth out of five segments
        part1_x = [x5 x4 x3 x1];
        part1_y = [y5 y4 y3 y1];
        part2_x = -part1_x;
        part2_y = part1_y;
        start_theta = -angle_tt_mr/2 + pi/2;
        end_theta = +angle_tt_mr/2 + pi/2;
        number_points = 20;
        resolution = (end_theta-start_theta)/number_points;
```

```
part3_theta = start_theta:resolution:end_theta;
part3_x = radius_si*cos(part3_theta);
part3_y = radius_si*sin(part3_theta);
ang_bet_slots = 2*pi/number_slots;
start_theta = pi/2 - ang_bet_slots/2;
end_theta = pi/2 - asin(x5/y6);
number_points = 20;
resolution = (end_theta-start_theta)/number_points;
part4_theta = start_theta:resolution:end_theta;
part4_x = y6*cos(part4_theta);
part4_y = y6*sin(part4_theta);
start_theta = pi/2 + asin(x5/y6);
end_theta = pi/2 + ang_bet_slots/2;
number_points = 20;
resolution = (end_theta-start_theta)/number_points;
part5_theta = start_theta:resolution:end_theta;
part5_x = y6*cos(part5_theta);
part5_y = y6*sin(part5_theta);

%draw the prototype tooth
plot(part1_x,part1_y);
plot(part2_x,part2_y);
plot(part3_x,part3_y);
plot(part4_x,part4_y);
plot(part5_x,part5_y);

%rotate tooth and draw
for j=2:number_slots
        rot_angle = (j-1)*ang_bet_slots;
        new_part1_x = part1_x*cos(rot_angle) - part1_y*sin(rot_angle);
        new_part1_y = part1_x*sin(rot_angle) + part1_y*cos(rot_angle);

        new_part2_x = part2_x*cos(rot_angle) - part2_y*sin(rot_angle);
        new_part2_y = part2_x*sin(rot_angle) + part2_y*cos(rot_angle);

        new_part3_x = part3_x*cos(rot_angle) - part3_y*sin(rot_angle);
        new_part3_y = part3_x*sin(rot_angle) + part3_y*cos(rot_angle);

        new_part4_x = part4_x*cos(rot_angle) - part4_y*sin(rot_angle);
        new_part4_y = part4_x*sin(rot_angle) + part4_y*cos(rot_angle);

        new_part5_x = part5_x*cos(rot_angle) - part5_y*sin(rot_angle);
        new_part5_y = part5_x*sin(rot_angle) + part5_y*cos(rot_angle);

        plot(new_part1_x,new_part1_y);
        plot(new_part2_x,new_part2_y);
        plot(new_part3_x,new_part3_y);
        plot(new_part4_x,new_part4_y);
        plot(new_part5_x,new_part5_y);
end


%form a prototype magnet out of three segments
angle_magnet = k_mag_ang * 2*pi/number_poles;

part1_x = [radius_ro*sin(angle_magnet/2) radius_ym*sin(angle_magnet/2)];
part1_y = [radius_ro*cos(angle_magnet/2) radius_ym*cos(angle_magnet/2)];
part2_x = -part1_x;
part2_y = part1_y;
start_theta = pi/2-angle_magnet/2;
end_theta = pi/2+angle_magnet/2;
number_points = 20;
resolution = (end_theta-start_theta)/number_points;
part3_theta = start_theta:resolution:end_theta;
part3_x = radius_ro*cos(part3_theta);
part3_y = radius_ro*sin(part3_theta);

%draw the prototype magnet
plot(part1_x,part1_y);
plot(part2_x,part2_y);
plot(part3_x,part3_y);

%rotate magnet and draw
ang_bet_magnets = 2*pi/number_poles;
for j=2:number_poles
        rot_angle = (j-1)*ang_bet_magnets;
        new_part1_x = part1_x*cos(rot_angle) - part1_y*sin(rot_angle);
```

```
            new_part1_y = part1_x*sin(rot_angle) + part1_y*cos(rot_angle);

            new_part2_x = part2_x*cos(rot_angle) - part2_y*sin(rot_angle);
            new_part2_y = part2_x*sin(rot_angle) + part2_y*cos(rot_angle);

            new_part3_x = part3_x*cos(rot_angle) - part3_y*sin(rot_angle);
            new_part3_y = part3_x*sin(rot_angle) + part3_y*cos(rot_angle);

            plot(new_part1_x,new_part1_y);
            plot(new_part2_x,new_part2_y);
            plot(new_part3_x,new_part3_y);
        end

        %draw the rotor circles
        number_points = 100;
        resolution = 2*pi/number_points;
        theta = 0:resolution:2*pi;
        circ_x = radius_ym*cos(theta);
        circ_y = radius_ym*sin(theta);
        plot(circ_x,circ_y);
        circ_x = radius_ri*cos(theta);
        circ_y = radius_ri*sin(theta);
        plot(circ_x,circ_y);


elseif (rotor_type == 1)      %outer rotor

        x1 = 0;                                                                 y1 =
radius_so;
        x2 = y1*sin(angle_tt_mr/2);                          y2=y1*cos(angle_tt_mr/2);
        x3 = (y1-depth_tt)*sin(angle_tt_mr/2);       y3=(y1-
depth_tt)*cos(angle_tt_mr/2);
            ang_1 = ang_tooth_tip_r - (pi/2) - (angle_tt_mr/2);
        x4 = width_t/2;                                                         y4 = y3 -
(x3 - x4)*tan(ang_1);
        x5 = 0;                                                                 y5 =
radius_si + width_sy;
        x6 = width_t/2;                                                         y6 =
sqrt(y5^2-x6^2);
        x7 = 0;                                                                 y7 =
radius_si;

        %draw the outside rotor radius and radius_ym
        hold off;
        number_points = 100;
        resolution = 2*pi/number_points;
        theta = 0:resolution:2*pi;
        circ_x = radius_ro*cos(theta);
        circ_y = radius_ro*sin(theta);
        plot(circ_x,circ_y);
        hold on;
        circ_x = radius_ym*cos(theta);
        circ_y = radius_ym*sin(theta);
        plot(circ_x,circ_y);

        %form a prototype magnet out of three segments
        angle_magnet = k_mag_ang * 2*pi/number_poles;

        part1_x = [radius_ri*sin(angle_magnet/2) radius_ym*sin(angle_magnet/2)];
        part1_y = [radius_ri*cos(angle_magnet/2) radius_ym*cos(angle_magnet/2)];
        part2_x = -part1_x;
        part2_y = part1_y;
        start_theta = pi/2-angle_magnet/2;
        end_theta = pi/2+angle_magnet/2;
        number_points = 20;
        resolution = (end_theta-start_theta)/number_points;
        part3_theta = start_theta:resolution:end_theta;
        part3_x = radius_ri*cos(part3_theta);
        part3_y = radius_ri*sin(part3_theta);

        %draw the prototype magnet
        plot(part1_x,part1_y);
        plot(part2_x,part2_y);
        plot(part3_x,part3_y);

        %rotate magnet and draw
        ang_bet_magnets = 2*pi/number_poles;
```

```
for j=2:number_poles
        rot_angle = (j-1)*ang_bet_magnets;
        new_part1_x = part1_x*cos(rot_angle) - part1_y*sin(rot_angle);
        new_part1_y = part1_x*sin(rot_angle) + part1_y*cos(rot_angle);

        new_part2_x = part2_x*cos(rot_angle) - part2_y*sin(rot_angle);
        new_part2_y = part2_x*sin(rot_angle) + part2_y*cos(rot_angle);

        new_part3_x = part3_x*cos(rot_angle) - part3_y*sin(rot_angle);
        new_part3_y = part3_x*sin(rot_angle) + part3_y*cos(rot_angle);

        plot(new_part1_x,new_part1_y);
        plot(new_part2_x,new_part2_y);
        plot(new_part3_x,new_part3_y);
end

%form a prototype tooth out of five segments
part1_x = [x6 x4 x3 x2];
part1_y = [y6 y4 y3 y2];
part2_x = -part1_x;
part2_y = part1_y;
start_theta = -angle_tt_mr/2 + pi/2;
end_theta = +angle_tt_mr/2 + pi/2;
number_points = 20;
resolution = (end_theta-start_theta)/number_points;
part3_theta = start_theta:resolution:end_theta;
part3_x = radius_so*cos(part3_theta);
part3_y = radius_so*sin(part3_theta);
ang_bet_slots = 2*pi/number_slots;
start_theta = pi/2 - ang_bet_slots/2;
end_theta = pi/2 - asin(x6/y5);
number_points = 20;
resolution = (end_theta-start_theta)/number_points;
part4_theta = start_theta:resolution:end_theta;
part4_x = y5*cos(part4_theta);
part4_y = y5*sin(part4_theta);
start_theta = pi/2 + asin(x6/y5);
end_theta = pi/2 + ang_bet_slots/2;
number_points = 20;
resolution = (end_theta-start_theta)/number_points;
part5_theta = start_theta:resolution:end_theta;
part5_x = y5*cos(part5_theta);
part5_y = y5*sin(part5_theta);

%draw the prototype tooth
plot(part1_x,part1_y);
plot(part2_x,part2_y);
plot(part3_x,part3_y);
plot(part4_x,part4_y);
plot(part5_x,part5_y);

%rotate tooth and draw
for j=2:number_slots
        rot_angle = (j-1)*ang_bet_slots;
        new_part1_x = part1_x*cos(rot_angle) - part1_y*sin(rot_angle);
        new_part1_y = part1_x*sin(rot_angle) + part1_y*cos(rot_angle);

        new_part2_x = part2_x*cos(rot_angle) - part2_y*sin(rot_angle);
        new_part2_y = part2_x*sin(rot_angle) + part2_y*cos(rot_angle);

        new_part3_x = part3_x*cos(rot_angle) - part3_y*sin(rot_angle);
        new_part3_y = part3_x*sin(rot_angle) + part3_y*cos(rot_angle);

        new_part4_x = part4_x*cos(rot_angle) - part4_y*sin(rot_angle);
        new_part4_y = part4_x*sin(rot_angle) + part4_y*cos(rot_angle);

        new_part5_x = part5_x*cos(rot_angle) - part5_y*sin(rot_angle);
        new_part5_y = part5_x*sin(rot_angle) + part5_y*cos(rot_angle);

        plot(new_part1_x,new_part1_y);
        plot(new_part2_x,new_part2_y);
        plot(new_part3_x,new_part3_y);
        plot(new_part4_x,new_part4_y);
        plot(new_part5_x,new_part5_y);
end

%draw the stator inside circle
```

```
        number_points = 100;
        resolution = 2*pi/number_points;
        theta = 0:resolution:2*pi;
        circ_x = radius_si*cos(theta);
        circ_y = radius_si*sin(theta);
        plot(circ_x,circ_y);

end
```

Generate the AutoCAD geometry function:

```
function generate_autocad(rotor_type, width_t, width_sy, width_ry, radius_si,
radius_so, radius_ri, radius_ro, radius_ym, angle_tt_mr, ang_tooth_tip_r,
number_slots, number_poles, k_mag_ang, depth_tt)
%start autocad file
[fidd,err] = DXF_start('motor_data.dxf',1.00);  % Unit scale is 1.00
if(err<0)
        return;
end

if (rotor_type == 0)   %inner rotor
        x2 = 0;                                         y2 = radius_si;
        x1 = y2*sin(angle_tt_mr/2);                     y1=y2*cos(angle_tt_mr/2);
        x3 = (y2+depth_tt)*sin(angle_tt_mr/2);
        y3=(y2+depth_tt)*cos(angle_tt_mr/2);
                ang_1 = pi - (angle_tt_mr/2) - ang_tooth_tip_r;
                ang_2 = (pi/2) - ang_1;
        x4 = width_t/2;                                 y4 = (x3 - x4)*tan(ang_2) + y3;
        x6 = 0;                                         y6 = radius_so - width_sy;
        x5 = width_t/2;                                 y5 = sqrt(y6^2-x5^2);
        x7 = 0;                                         y7 = radius_so;


        %draw the outside stator radius
        DXF_circle(fidd,0,0,radius_so,5,1);

        %form a prototype tooth out of five segments
        lines_part1_x = [x5 x4 x3 x1];
        lines_part1_y = [y5 y4 y3 y1];

        lines_part2_x = -lines_part1_x;
        lines_part2_y = lines_part1_y;

        arc_part3_radius = radius_si;
        arc_part3_start_angle = (-angle_tt_mr/2 + pi/2)*180/pi;
        arc_part3_end_angle = (+angle_tt_mr/2 + pi/2)*180/pi;


        ang_bet_slots = 2*pi/number_slots;
        arc_part4_radius = y6;
        arc_part4_start_angle = (pi/2 - ang_bet_slots/2)*180/pi;
        arc_part4_end_angle = (pi/2 - asin(x5/y6))*180/pi;


        arc_part5_radius = y6;
        arc_part5_start_angle = (pi/2 + asin(x5/y6))*180/pi;
        arc_part5_end_angle = (pi/2 + ang_bet_slots/2)*180/pi;

        %form prottype boundary for conductors
        arc_part6_radius = sqrt(x3^2+y3^2);
        arc_part6_start_angle = (pi/2 - ang_bet_slots/2)*180/pi;
        arc_part6_end_angle = (pi/2 - atan(x3/y3))*180/pi;

        arc_part7_radius = sqrt(x3^2+y3^2);
        arc_part7_start_angle = (pi/2 + atan(x3/y3))*180/pi;
        arc_part7_end_angle = (pi/2 + ang_bet_slots/2)*180/pi;

        %form lines for intersection of coils if there are two coils per slot
        lines_part8_x = [sqrt(x3^2+y3^2)*sin(ang_bet_slots/2)
y6*sin(ang_bet_slots/2)];
        lines_part8_y = [sqrt(x3^2+y3^2)*cos(ang_bet_slots/2)
y6*cos(ang_bet_slots/2)];

        lines_part9_x = -lines_part8_x;
```

```
        lines_part9_y = lines_part8_y;

        %rotate tooth and draw
        for j=1:number_slots

                rot_angle = (j-1)*ang_bet_slots;
                new_lines_part1_x = lines_part1_x*cos(rot_angle) -
lines_part1_y*sin(rot_angle);
                new_lines_part1_y = lines_part1_x*sin(rot_angle) +
lines_part1_y*cos(rot_angle);
                new_lines_part2_x = lines_part2_x*cos(rot_angle) -
lines_part2_y*sin(rot_angle);
                new_lines_part2_y = lines_part2_x*sin(rot_angle) +
lines_part2_y*cos(rot_angle);

                new_arc_part3_start_angle = arc_part3_start_angle + rot_angle*180/pi;
                new_arc_part3_end_angle = arc_part3_end_angle + rot_angle*180/pi;

                new_arc_part4_start_angle = arc_part4_start_angle + rot_angle*180/pi;
                new_arc_part4_end_angle = arc_part4_end_angle + rot_angle*180/pi;

                new_arc_part5_start_angle = arc_part5_start_angle + rot_angle*180/pi;
                new_arc_part5_end_angle = arc_part5_end_angle + rot_angle*180/pi;

                new_arc_part6_start_angle = arc_part6_start_angle + rot_angle*180/pi;
                new_arc_part6_end_angle = arc_part6_end_angle + rot_angle*180/pi;

                new_arc_part7_start_angle = arc_part7_start_angle + rot_angle*180/pi;
                new_arc_part7_end_angle = arc_part7_end_angle + rot_angle*180/pi;

                new_lines_part8_x = lines_part8_x*cos(rot_angle) -
lines_part8_y*sin(rot_angle);
                new_lines_part8_y = lines_part8_x*sin(rot_angle) +
lines_part8_y*cos(rot_angle);
                new_lines_part9_x = lines_part9_x*cos(rot_angle) -
lines_part9_y*sin(rot_angle);
                new_lines_part9_y = lines_part9_x*sin(rot_angle) +
lines_part9_y*cos(rot_angle);


        DXF_line(fidd,new_lines_part1_x(1),new_lines_part1_y(1),new_lines_part1_x(2),n
ew_lines_part1_y(2),5,1);

        DXF_line(fidd,new_lines_part1_x(2),new_lines_part1_y(2),new_lines_part1_x(3),n
ew_lines_part1_y(3),5,1);

        DXF_line(fidd,new_lines_part1_x(3),new_lines_part1_y(3),new_lines_part1_x(4),n
ew_lines_part1_y(4),5,1);

        DXF_line(fidd,new_lines_part2_x(1),new_lines_part2_y(1),new_lines_part2_x(2),n
ew_lines_part2_y(2),5,1);

        DXF_line(fidd,new_lines_part2_x(2),new_lines_part2_y(2),new_lines_part2_x(3),n
ew_lines_part2_y(3),5,1);

        DXF_line(fidd,new_lines_part2_x(3),new_lines_part2_y(3),new_lines_part2_x(4),n
ew_lines_part2_y(4),5,1);

        DXF_arc(fidd,0,0,arc_part3_radius,new_arc_part3_start_angle,new_arc_part3_end_
angle,5,1);

        DXF_arc(fidd,0,0,arc_part4_radius,new_arc_part4_start_angle,new_arc_part4_end_
angle,5,1);

        DXF_arc(fidd,0,0,arc_part5_radius,new_arc_part5_start_angle,new_arc_part5_end_
angle,5,1);

        DXF_arc(fidd,0,0,arc_part6_radius,new_arc_part6_start_angle,new_arc_part6_end_
angle,5,1);

        DXF_arc(fidd,0,0,arc_part7_radius,new_arc_part7_start_angle,new_arc_part7_end_
angle,5,1);

        DXF_line(fidd,new_lines_part8_x(1),new_lines_part8_y(1),new_lines_part8_x(2),n
ew_lines_part8_y(2),5,1);
```

```
        DXF_line(fidd,new_lines_part9_x(1),new_lines_part9_y(1),new_lines_part9_x(2),n
ew_lines_part9_y(2),5,1);

        end


        % %form a prototype magnet out of three segments
        angle_magnet = k_mag_ang * 2*pi/number_poles;

        part1_x = [radius_ro*sin(angle_magnet/2) radius_ym*sin(angle_magnet/2)];
        part1_y = [radius_ro*cos(angle_magnet/2) radius_ym*cos(angle_magnet/2)];

        part2_x = -part1_x;
        part2_y = part1_y;

        part3_start_theta = (pi/2-angle_magnet/2)*180/pi;
        part3_end_theta = (pi/2+angle_magnet/2)*180/pi;
        part3_radius = radius_ro;

        % %rotate magnet and draw
        ang_bet_magnets = 2*pi/number_poles;
        for j=1:number_poles
                rot_angle = (j-1)*ang_bet_magnets;
                new_part1_x = part1_x*cos(rot_angle) - part1_y*sin(rot_angle);
                new_part1_y = part1_x*sin(rot_angle) + part1_y*cos(rot_angle);

                new_part2_x = part2_x*cos(rot_angle) - part2_y*sin(rot_angle);
                new_part2_y = part2_x*sin(rot_angle) + part2_y*cos(rot_angle);

                new_part3_start_theta = part3_start_theta+rot_angle*180/pi;
                new_part3_end_theta = part3_end_theta+rot_angle*180/pi;

        DXF_line(fidd,new_part1_x(1),new_part1_y(1),new_part1_x(2),new_part1_y(2),5,1)
        ;
        DXF_line(fidd,new_part2_x(1),new_part2_y(1),new_part2_x(2),new_part2_y(2),5,1)
        ;
        DXF_arc(fidd,0,0,part3_radius,new_part3_start_theta,new_part3_end_theta,5,1);
        end

        % %draw the rotor circles
        DXF_circle(fidd,0,0,radius_ym,5,1);
        DXF_circle(fidd,0,0,radius_ri,5,1);




elseif (rotor_type == 1)      %outer rotor

        x1 = 0;                                 y1 = radius_so;
        x2 = y1*sin(angle_tt_mr/2);             y2=y1*cos(angle_tt_mr/2);
        x3 = (y1-depth_tt)*sin(angle_tt_mr/2);y3=(y1-depth_tt)*cos(angle_tt_mr/2);
                ang_1 = ang_tooth_tip_r - (pi/2) - (angle_tt_mr/2);
        x4 = width_t/2;                         y4 = y3 - (x3 - x4)*tan(ang_1);
        x5 = 0;                                 y5 = radius_si + width_sy;
        x6 = width_t/2;                         y6 = sqrt(y5^2-x6^2);
        x7 = 0;                                 y7 = radius_si;

        %draw the inside stator radius
        DXF_circle(fidd,0,0,radius_si,5,1);

        %form a prototype tooth out of five segments
        lines_part1_x = [x2 x3 x4 x6];
        lines_part1_y = [y2 y3 y4 y6];

        lines_part2_x = -lines_part1_x;
        lines_part2_y = lines_part1_y;

        arc_part3_radius = radius_so;
        arc_part3_start_angle = (-angle_tt_mr/2 + pi/2)*180/pi;
        arc_part3_end_angle = (+angle_tt_mr/2 + pi/2)*180/pi;


        ang_bet_slots = 2*pi/number_slots;
        arc_part4_radius = y5;
        arc_part4_start_angle = (pi/2 - ang_bet_slots/2)*180/pi;
```

```
        arc_part4_end_angle = (pi/2 - asin(x6/y5))*180/pi;


        arc_part5_radius = y5;
        arc_part5_start_angle = (pi/2 + asin(x6/y5))*180/pi;
        arc_part5_end_angle = (pi/2 + ang_bet_slots/2)*180/pi;

        %form prottype boundary for conductors
        arc_part6_radius = sqrt(x3^2+y3^2);
        arc_part6_start_angle = (pi/2 - ang_bet_slots/2)*180/pi;
        arc_part6_end_angle = (pi/2 - atan(x3/y3))*180/pi;

        arc_part7_radius = sqrt(x3^2+y3^2);
        arc_part7_start_angle = (pi/2 + atan(x3/y3))*180/pi;
        arc_part7_end_angle = (pi/2 + ang_bet_slots/2)*180/pi;

        %form lines for intersection of coils if there are two coils per slot
        lines_part8_x = [sqrt(x3^2+y3^2)*sin(ang_bet_slots/2)
y5*sin(ang_bet_slots/2)];
        lines_part8_y = [sqrt(x3^2+y3^2)*cos(ang_bet_slots/2)
y5*cos(ang_bet_slots/2)];

        lines_part9_x = -lines_part8_x;
        lines_part9_y = lines_part8_y;

        %rotate tooth and draw
        for j=1:number_slots

                rot_angle = (j-1)*ang_bet_slots;
                new_lines_part1_x = lines_part1_x*cos(rot_angle) -
lines_part1_y*sin(rot_angle);
                new_lines_part1_y = lines_part1_x*sin(rot_angle) +
lines_part1_y*cos(rot_angle);
                new_lines_part2_x = lines_part2_x*cos(rot_angle) -
lines_part2_y*sin(rot_angle);
                new_lines_part2_y = lines_part2_x*sin(rot_angle) +
lines_part2_y*cos(rot_angle);

                new_arc_part3_start_angle = arc_part3_start_angle + rot_angle*180/pi;
                new_arc_part3_end_angle = arc_part3_end_angle + rot_angle*180/pi;

                new_arc_part4_start_angle = arc_part4_start_angle + rot_angle*180/pi;
                new_arc_part4_end_angle = arc_part4_end_angle + rot_angle*180/pi;

                new_arc_part5_start_angle = arc_part5_start_angle + rot_angle*180/pi;
                new_arc_part5_end_angle = arc_part5_end_angle + rot_angle*180/pi;

                new_arc_part6_start_angle = arc_part6_start_angle + rot_angle*180/pi;
                new_arc_part6_end_angle = arc_part6_end_angle + rot_angle*180/pi;

                new_arc_part7_start_angle = arc_part7_start_angle + rot_angle*180/pi;
                new_arc_part7_end_angle = arc_part7_end_angle + rot_angle*180/pi;

                new_lines_part8_x = lines_part8_x*cos(rot_angle) -
lines_part8_y*sin(rot_angle);
                new_lines_part8_y = lines_part8_x*sin(rot_angle) +
lines_part8_y*cos(rot_angle);
                new_lines_part9_x = lines_part9_x*cos(rot_angle) -
lines_part9_y*sin(rot_angle);
                new_lines_part9_y = lines_part9_x*sin(rot_angle) +
lines_part9_y*cos(rot_angle);


        DXF_line(fidd,new_lines_part1_x(1),new_lines_part1_y(1),new_lines_part1_x(2),n
ew_lines_part1_y(2),5,1);

        DXF_line(fidd,new_lines_part1_x(2),new_lines_part1_y(2),new_lines_part1_x(3),n
ew_lines_part1_y(3),5,1);

        DXF_line(fidd,new_lines_part1_x(3),new_lines_part1_y(3),new_lines_part1_x(4),n
ew_lines_part1_y(4),5,1);

        DXF_line(fidd,new_lines_part2_x(1),new_lines_part2_y(1),new_lines_part2_x(2),n
ew_lines_part2_y(2),5,1);

        DXF_line(fidd,new_lines_part2_x(2),new_lines_part2_y(2),new_lines_part2_x(3),n
ew_lines_part2_y(3),5,1);
```

```
        DXF_line(fidd,new_lines_part2_x(3),new_lines_part2_y(3),new_lines_part2_x(4),n
ew_lines_part2_y(4),5,1);


        DXF_arc(fidd,0,0,arc_part3_radius,new_arc_part3_start_angle,new_arc_part3_end_
angle,5,1);

        DXF_arc(fidd,0,0,arc_part4_radius,new_arc_part4_start_angle,new_arc_part4_end_
angle,5,1);

        DXF_arc(fidd,0,0,arc_part5_radius,new_arc_part5_start_angle,new_arc_part5_end_
angle,5,1);

        DXF_arc(fidd,0,0,arc_part6_radius,new_arc_part6_start_angle,new_arc_part6_end_
angle,5,1);

        DXF_arc(fidd,0,0,arc_part7_radius,new_arc_part7_start_angle,new_arc_part7_end_
angle,5,1);


        DXF_line(fidd,new_lines_part8_x(1),new_lines_part8_y(1),new_lines_part8_x(2),n
ew_lines_part8_y(2),5,1);

        DXF_line(fidd,new_lines_part9_x(1),new_lines_part9_y(1),new_lines_part9_x(2),n
ew_lines_part9_y(2),5,1);

        end


        % %form a prototype magnet out of three segments
        angle_magnet = k_mag_ang * 2*pi/number_poles;

        part1_x = [radius_ri*sin(angle_magnet/2) radius_ym*sin(angle_magnet/2)];
        part1_y = [radius_ri*cos(angle_magnet/2) radius_ym*cos(angle_magnet/2)];

        part2_x = -part1_x;
        part2_y = part1_y;

        part3_start_theta = (pi/2-angle_magnet/2)*180/pi;
        part3_end_theta = (pi/2+angle_magnet/2)*180/pi;
        part3_radius = radius_ri;

        % %rotate magnet and draw
        ang_bet_magnets = 2*pi/number_poles;
        for j=1:number_poles
                rot_angle = (j-1)*ang_bet_magnets;
                new_part1_x = part1_x*cos(rot_angle) - part1_y*sin(rot_angle);
                new_part1_y = part1_x*sin(rot_angle) + part1_y*cos(rot_angle);

                new_part2_x = part2_x*cos(rot_angle) - part2_y*sin(rot_angle);
                new_part2_y = part2_x*sin(rot_angle) + part2_y*cos(rot_angle);

                new_part3_start_theta = part3_start_theta+rot_angle*180/pi;
                new_part3_end_theta = part3_end_theta+rot_angle*180/pi;


        DXF_line(fidd,new_part1_x(1),new_part1_y(1),new_part1_x(2),new_part1_y(2),5,1)
;

        DXF_line(fidd,new_part2_x(1),new_part2_y(1),new_part2_x(2),new_part2_y(2),5,1)
;

        DXF_arc(fidd,0,0,part3_radius,new_part3_start_theta,new_part3_end_theta,5,1);
        end

        % %draw the rotor circles
        DXF_circle(fidd,0,0,radius_ym,5,1);
        DXF_circle(fidd,0,0,radius_ro,5,1);

end


DXF_end(fidd);
```